Universidad
Carlos III de Madrid

CALIDAD Y SEGURIDAD A NIVEL DE FILAS EN BBDD ORACLE

PROYECTO FIN DE CARRERA

Ingeniería Técnica en Informática de Gestión

Autor: Javier Fernández Cuéllar **NIA:** 100039895

Tutor: Miguel Ángel Ramos

Septiembre 2009

Agradecimientos

En primer lugar quiero agradecer a mi familia los valores que me han inculcado desde pequeño para llegar a ser la persona que soy hoy día. Por vuestro apoyo durante todos estos años y los que quedan...

A todas esas personas que he conocido en la universidad y que con el tiempo nos hemos hecho grandes amigos, Mari, Óscar, Paula y Marisa. Gracias por todos los buenos momentos que hemos vivido y los que nos quedan por vivir…

A mis compañeros de Oesía que están ubicados en ICM, Domingo, Manu y Sebas, porque hasta en los malos momentos entre todos somos capaces de sacar algo bueno con lo que poder sonreír y hacer nuestro trabajo más ameno.

A mis antiguos compañeros de Oesía, Laura, Obdulia, Javi y Dani por los buenos momentos que he vivido con vosotros en mi iniciación en el mundo laboral.

Finalmente quiero agradecer a Miguel Ángel Ramos toda la ayuda que me ha proporcionado. Gracias por la atención, ánimos, disponibilidad y consejos dados durante todo este tiempo.

ÍNDICE

ÍNDICE DE TABLAS

ÍNDICE DE IMÁGENES

Introducción

Actualmente, obtener o proporcionar productos de calidad en cualquier ámbito es fundamental para una organización.

En el mundo de la informática, hará unas cuantas décadas, más o menos, que los componentes software se han ido introduciendo paulatinamente en nuestras vidas, hasta tal extremo que hoy en día no hay apenas elementos que no dispongan de un software para su funcionamiento. Desde un simple software que permite el buen funcionamiento de un microondas, un teléfono móvil... hasta el software de los instrumentos de navegación que hay en los aviones, el software de los satélites, e incluso el software que tienen máquinas que permiten controlar las constantes vitales de un enfermo.

Para asegurar el buen funcionamiento del software bajo cualquier circunstancia, es necesario establecer una serie de pautas para hacer bien tanto los desarrollos de un nuevo software, como el mantenimiento de un software ya elaborado. De manera que el producto software se encuentre probado, antes de pasar al estado de producción, y así asegurar la calidad del mismo. Además, mediante la implantación de esta serie de pautas (modelos y metodologías), para asegurar la calidad del software, la organización puede obtener certificaciones emitidas por organismos de carácter internacional. Estas certificaciones servirán como demostración de la excelencia de los procesos productivos que lleva a cabo la organización, lo que captará la atención de nuevos clientes y el mantenimiento de los existentes.

En la evolución experimentada a lo largo de los años en la calidad del software, se ha pasado de un tratamiento centrado en la inspección y detección de errores, a una aproximación más sistemática, dada la importancia que ha adquirido la calidad en la ingeniería del software.

En las técnicas informáticas para el almacenamiento de datos, en nuestro caso las bases de datos, se puede comprobar cómo aparte de las pautas que deben seguirse para obtener un producto de calidad, debe tenerse en cuenta, que el activo más importante es la seguridad, conservación y privacidad de los datos bajo cualquier circunstancia, que estos estén disponibles en cualquier momento y que después de hacer cualquier tipo de operación sobre los datos, la base de datos quede en un estado de coherencia.

Hoy en día, ORACLE proporciona una de las herramientas más completas del mercado para poder implementar y satisfacer todos los requisitos necesarios. ORACLE surge a finales de los 70 bajo el nombre de Relational Software a partir de un estudio sobre SGBD (Sistemas Gestores de Base de Datos) de George Koch. Computer World definió este estudio como uno de los más completos jamás escritos sobre bases de datos. Este artículo incluía una comparativa de productos que erigía a Relational Software como el más completo desde el punto de vista técnico. Esto se debía a que usaba la filosofía de las bases de datos relacionales, algo que por aquella época era todavía desconocido.

El objetivo principal de este estudio teórico, junto con ejemplos prácticos, se divide en dos. El primero es dar a conocer las pautas esenciales para obtener un producto software de calidad en una base de datos. Para ello es necesario estudiar los diferentes aspectos relacionados con el concepto del término calidad en cuanto al software.

El segundo objetivo, dado que la calidad está muy ligada a la seguridad proporcionada a los datos dentro de una base de datos, es exponer los diferentes puntos de control a tener en cuenta y adentrarse en la seguridad a nivel de fila que proporciona la herramienta Oracle Label Security.

A continuación, se especifica la distribución del Proyecto Fin de Carrera, resumiendo cada una de las partes que constituyen el mismo.

Capítulo 1: Concepto de calidad

El objetivo de este primer capítulo es dar al lector una definición del concepto de calidad en el ámbito del software, características generales del software, principios de gestión de la calidad, así como una introducción a la historia y evolución de la calidad.

Capítulo 2: Aseguramiento de la calidad

Basándose en las normas ISO 9001 e ISO 90003, se exponen los diferentes requisitos que deben considerarse necesarios cumplir para obtener como resultado un producto software de calidad.

Capítulo 3: Introducción a Oracle

Este capítulo se divide en dos secciones. En la primera, se realiza una descripción introductoria a la definición de base de dados y las personas que interactúan con la misma. En la segunda sección, se expone que es un sistema gestor de base de datos, sus componentes, estructura, que ventajas e inconvenientes tiene la utilización de un sistema gestor de base de datos y la historia del sistema gestor de base de datos Oracle.

Capítulo 4: Calidad y seguridad en una base de datos Oracle

El fin de este capítulo es presentar los diferentes puntos de control a tener en cuenta en una base de datos Oracle, además de indicar al usuario como gestionar la seguridad. Se realiza una pequeña introducción a la base de datos virtual privada y a Oracle Label Security.

Capítulo 5: Proceso de instalación de Oracle 11g

Para poder utilizar la herramienta Oracle Label Security, debe llevarse a cabo una configuración concreta de los parámetros de instalación. Este capítulo muestra los pasos a seguir para realizar la instalación de Oracle 11g junto con la herramienta Oracle Label Security.

Capítulo 6: Nociones básicas de Oracle Label Security

Llegado a este punto, se exponen los principios en los que se basa Oracle Label Security, los pasos a seguir para implantar correctamente una política Oracle Label Security, con el objetivo de enmascarar los datos a los que el usuario no debe tener acceso, y la implementación de un caso práctico.

Capítulo 7: Uso de Oracle Label Security para gestionar el etiquetado de datos

Avanzando en el uso de Oracle Label Security, en este punto se muestran los conceptos básicos para implementar una política aplicada a una tabla de base de datos, con el fin de conseguir seguridad a nivel de fila en dicha tabla, y el desarrollo de un caso práctico.

Capítulo 8: Auditoría en Oracle Label Security

En este último capítulo, se exponen las nociones básicas para llevar a cabo la activación y seguimiento de una auditoría sobre una política Oracle Label Security. Además, se muestra un pequeño caso práctico.

Anexo I: Gestión de autorizaciones de usuario

Anexo en el que se muestran las características de las autorizaciones de etiquetas de usuario y las diferentes autorizaciones especiales que se pueden otorgar a un usuario.

Anexo II: Administración de etiquetas de usuario y privilegios

Anexo en el que se presenta al usuario los diferentes procedimientos para gestionar las etiquetas de usuario de una política Oracle Label Security.

Anexo III: Opciones para la aplicación de políticas Oracle Label Security

Anexo en el que se exponen las diferentes opciones que hay para aplicar una política Oracle Label Security.

Capítulo 1: Concepto de calidad

Definición

Inicialmente con el objetivo de estudiar la gestión de la calidad en los productos software es necesario definir anteriormente este atributo. Hay decenas de definiciones aportadas por diferentes medios (instituciones, estudiosos y organizaciones).

El concepto de *"calidad"* es definido por la Real Academia Española de la Lengua como:

Calidad

1. Conjunto de cualidades que constituyen la manera de ser de una persona o cosa: tela de superior ~; persona de noble ~ y honradez a toda prueba; ~ de vida, nivel de bienestar de los individuos de una sociedad.

2. Superioridad en su línea; nobleza de linaje; importancia o gravedad de alguna cosa: una mercancía de ~; una dama de ~, el que tiene más valor, por la condición del que lo emite. Superioridad o excelencia: La calidad del vino de Jerez ha conquistado los mercados.

3. Consideración social, civil o política; circunstancias personales de un individuo en relación con algún empleo o dignidad: ~ de ciudadano; en ~ de, con el carácter o la investidura de. Clase, condición: nos atendió en calidad de abogado de la familia.

4. Condición o requisito que se pone en un contrato: a ~ de que, loc. Conj. Con la condición de que.

5. Sensación de realidad táctil de cualquier materia representada en una pintura: el valor tonal de un color, dependiente de la pureza de la luz.

6. Prendas morales.

7. Propiedad o conjunto de propiedades inherentes a algo, que permiten juzgar su valor: Esta tela es de buena calidad.

8. Nobleza de linaje: a la recepción sólo asistieron personas de calidad.

9. Importancia: fijes una obra de calidad.

Otras definiciones de calidad son:

"Calidad es el grado en el que el conjunto de características inherentes cumple con los requisitos". (Norma ISO 9000)

"La calidad como resultado de la interacción de dos dimensiones: dimensión subjetiva (lo que el cliente quiere) y dimensión objetiva (lo que se ofrece)". (Shewart)

"Calidad es la menor pérdida posible para la sociedad". (Taguchi)

Calidad en relación al software

El software conlleva una serie de especificidades con relación a la calidad. Detallando lo que se entiende por calidad de software se tendría que hablar de tres niveles diferenciados, los cuales se muestran en la siguiente imagen:

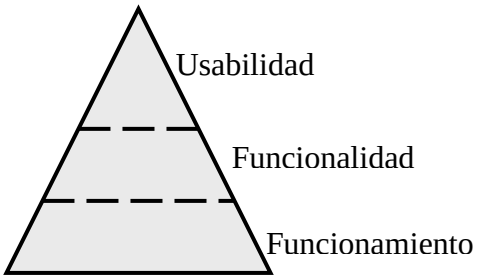

Imagen 1. Niveles de la calidad

Funcionamiento: Sería el nivel más bajo, asumido. El software debe funcionar siempre, en todo momento; debe permitir ser utilizado cuando sea necesario.

Funcionalidad: Sería el nivel intermedio. El software deberá cubrir las funcionalidades que publica; debe hacer lo que dice que hace.

Usabilidad: Sería el nivel superior. No sólo un software debe hacer lo que dice que hace; también debe permitir hacerlo de forma adecuada, natural.

En general, con respecto al software, se quiere que funcione siempre, que tenga las cualidades que dice tener y que se puedan utilizar todas las funcionalidades de una forma natural.

Características del software

La calidad del producto software se diferencia de la calidad de otros productos de fabricación industrial, ya que el software tiene sus propias características:

- **El software es un producto mental**, no restringido por las leyes de la física o por los límites de los procesos de fabricación. Es abstracto, intangible.

- **Se desarrolla, no se fabrica**. El coste está fundamentalmente en el proceso de diseño, no en la posterior producción en serie.

- **Los costes de desarrollo de software se concentran en las tareas de ingeniería**, mientras que en la fabricación clásica los costes se acentúan más en las tareas de producción.

- **El software no se deteriora con el paso del tiempo**. No es susceptible de los efectos del entorno y su curva de fallos es diferente de la del hardware. Cualquier problema que pueda surgir en la fase de mantenimiento, eran problemas que ya se encontraban allí desde un principio.

- **Es artesanal en gran medida**. El software, en su mayoría, se construye a medida lo que dificulta aún más el control de su calidad.

- **El mantenimiento del software es más complejo** que el mantenimiento del hardware. Cuando un componente hardware se deteriora se sustituye por otro, sin embargo cada fallo software implica un error de diseño o de programación.

- **El desarrollo de software es joven**, por lo que las técnicas de las que se dispone no están perfeccionadas.

Tipos de calidad

Existen tres tipos de calidad distintos, todos ellos deben estar relacionados. El objetivo de la gestión de la calidad es que la relación entre las tres sea la mayor posible. Estos tipos de calidad son:

- **Calidad necesaria:** Es la calidad que el cliente exige que tenga el producto o servicio que necesita y por tanto la que espera recibir. El cliente por lo general exigirá la máxima calidad dentro de un precio que él considere razonable.

- **Calidad programada:** Es el nivel de calidad que las organizaciones, que fabrican y ofrecen un servicio, se proponen obtener u ofrecer a sus clientes. En algunos casos los productos que se obtienen tienen una calidad que difiere considerablemente de la que se propusieron en un principio.

- **Calidad realizada:** Es la calidad que tiene finalmente el producto o servicio proporcionado a un cliente. Este tipo de calidad dependerá de diversos factores: experiencia de las personas que realizan el trabajo, tecnologías disponibles, conocimiento de las herramientas, etc.

Costes de calidad

A pesar de que la calidad ofrece una disminución de los costes para una organización a medio o largo plazo, la implantación de un sistema de calidad conlleva unos costes a corto plazo. Estos costes a corto plazo son:

- **Costes de prevención:** Son los costes que la organización deberá asumir para evitar que se produzcan errores en cualquier función de la empresa: administrativa, personal, codificación, análisis, etc.

- **Costes de evaluación:** Son los costes en los que se incurre para realizar la inspección de la producción una vez terminada y para que se produzcan auditorías sobre la conformidad de las funciones con procedimientos anteriormente indicados o seleccionados.

- **Costes de la no calidad:** Son los costes a los que se enfrenta una organización cuando se producen errores en la producción de un proyecto. Estos costes pueden ser tangibles o económicos, por ejemplo: coste de mantenimiento; pero también pueden ser intangibles, por ejemplo: la pérdida de confianza por parte de un cliente.

Principios de la gestión de la calidad

Con el objetivo de contribuir al proceso de mejora continuo de las organizaciones, a continuación se exponen los ocho principios de la gestión de la calidad sobre las cuales se basan las normas de sistemas de gestión de la calidad de la serie ISO 9000.

Los ocho principios de la gestión de la calidad están definidos en la norma ISO 9000 – *Sistemas de Gestión de la Calidad* – y en la norma ISO 9004 – *Sistemas de Gestión de la Calidad* –. Estos principios los puede utilizar la dirección de la organización como un marco de referencia para guiar a sus organizaciones en la consecución de la mejora del desempeño.

Con el fin de conducir y administrar una organización de forma exitosa, se requiere que ésta se dirija y controle de forma sistemática y transparente. Se puede lograr el éxito implementando y manteniendo un sistema de gestión que esté diseñado para mejorar continuamente su desempeño, mediante la consideración de las necesidades de todas las partes interesadas. La gestión de una organización comprende la gestión de la calidad entre otras disciplinas de gestión.

El propósito de una organización es: identificar y satisfacer las necesidades y expectativas de sus clientes y otras partes interesadas (empleados, proveedores, accionistas) para lograr ventaja competitiva y hacerlo de una manera eficaz y eficiente.

La aplicación de los principios de la gestión de la calidad no sólo proporciona beneficios directos, sino que también hace una importante contribución a la gestión de costos y riesgos.

Las consideraciones de beneficios, costos y gestión de riesgos, son importantes para la organización, sus clientes y otras partes interesadas. Estas consideraciones, en relación con el desempeño global de la organización, pueden tener impacto sobre:

- la fidelidad del cliente,
- la reiteración de negocios y referencia o recomendación de la organización,
- respuestas rápidas y flexibles a las oportunidades del mercado,
- costos y tiempos de ciclos mediante el uso eficaz y eficiente de los recursos,
- ventaja competitiva mediante capacidades mejoradas de la organización,
- comprensión y motivación de las personas hacia las metas y objetivos de la organización, así como participación en la mejora continua.

A pesar de que cada principio tiene utilidad por sí solo, es conveniente que se apliquen de forma integral, como un todo, donde existe una relación causa efecto entre los ocho principios, con el propósito de satisfacer las necesidades del cliente y cumplir el propósito de la organización.

Principio 1: Enfoque al cliente

Las organizaciones dependen de sus clientes y por lo tanto deberían comprender las necesidades actuales y futuras de los clientes, satisfacer los requisitos de los clientes y esforzarse en exceder las expectativas de los clientes.

La dirección debería establecer una organización orientada al cliente, mediante la definición de sistemas y procesos claramente comprensibles, gestionables y mejorables, en lo que a eficacia y eficiencia se refiere, y asegurándose de una eficaz y eficiente operación y control de los procesos, así como de las medidas y datos utilizados para determinar el desempeño satisfactorio de la organización.

Ejemplos de actividades útiles para establecer una organización orientada al cliente son:

a) definir y promover procesos que lleven a mejorar el desempeño de la organización,

b) adquirir y utilizar información y datos de los clientes de manera continua,

c) dirigir el progreso hacia la mejora continua de la satisfacción del cliente.

Beneficios clave:

- Aumento de los ingresos y de la porción del mercado, obtenido mediante respuestas rápidas y flexibles a las oportunidades del mercado.

- Aumento de la eficacia en el uso de los recursos de la organización para aumentar la satisfacción del cliente.

- Aumenta la fidelidad del cliente, lo cual lleva a reiterar tratos comerciales.

La aplicación del principio de enfoque al cliente conduce a lo siguiente:

- Investigar y comprender las necesidades y las expectativas del cliente.

- Asegurar que los objetivos de la organización están vinculados con las necesidades y expectativas del cliente.

- Comunicar las necesidades y las expectativas del cliente a toda la organización.

- Medir la satisfacción del cliente y actuar en base a los resultados.

- Gestionar sistemáticamente las relaciones con los clientes.

- Asegurar el equilibrio entre satisfacer a los clientes y a otras partes interesadas.

Principio 2: Liderazgo

Los líderes establecen la unidad de propósito y la orientación de la organización. Ellos deberían crear y mantener un ambiente interno en el cual el personal pueda llegar a involucrarse totalmente en el logro de los objetivos de la organización.

El liderazgo, compromiso y la participación activa de la dirección de la organización, son esenciales para desarrollar y mantener un sistema de gestión de la calidad eficaz y eficiente para que todas las partes interesadas logren beneficios. Para alcanzar estos beneficios es necesario establecer, mantener y aumentar la satisfacción del cliente. Algunas acciones que deberían ser consideradas son:

a) establecer una visión, políticas y objetivos estratégicos coherentes con el propósito de la institución,

b) participar en proyectos de mejora en la búsqueda de nuevos métodos, soluciones y servicios,

c) obtener directamente retroalimentación sobre la eficacia y eficiencia del sistema de gestión de la calidad.

Beneficios clave:

- Las personas comprenderán y se sentirán motivadas respecto de las metas de la organización.

- Las actividades son evaluadas, alineadas e implementadas en una manera unificada.

- Disminuirá la comunicación deficiente entre los distintos niveles de la empresa.

La aplicación del principio de liderazgo conduce a lo siguiente:

- Considerar las necesidades de todas las partes interesadas incluyendo clientes, propietarios, proveedores, accionistas, comunidades locales y la sociedad en su conjunto.

- Establecer una visión clara del futuro de la organización.

- Establecer metas y objetivos desafiantes.

- Crear y mantener valores compartidos, transparencia y modelos éticos en todos los niveles de la organización.

- Establecer confianza y eliminar los temores.

- Proporcionar a las personas los recursos necesarios, capacitación y libertad para actuar con responsabilidad.

Principio 3: Participación del personal

El personal, a todos los niveles, es la esencia de una organización y su total compromiso posibilita que sus habilidades sean usadas para el beneficio de la organización.

La dirección debería mejorar tanto la eficacia como la eficiencia de la organización, incluyendo el sistema de gestión de la calidad, mediante, la participación y el apoyo de las personas. Como ayuda en el logro de sus objetivos de mejora del desempeño, la dirección debería promover la participación y el desarrollo de su personal, por ejemplo:

- proporcionando formación continua y la planificación de carrera,
- definiendo sus responsabilidades y autoridades,
- mediante reconocimientos y recompensas.

Además, debería asegurarse de que se dispone de la competencia necesaria para la operación eficaz y eficiente de la organización. La dirección debería considerar el análisis tanto de las necesidades de competencia presentes como de las esperadas en comparación con la competencia ya existente en la organización.

Beneficios clave:

- Motivación, compromiso y participación de la gente en la organización.
- Innovación y creatividad en la persecución de los objetivos de la organización.
- Responsabilidad de los individuos respecto de su propio desempeño.
- Disposición de los individuos a participar en contribuir a la mejora continua.

La aplicación del principio de participación del personal conduce a que sus integrantes:

- Comprendan la importancia de su contribución y función en la organización.
- Identifiquen las restricciones de su desempeño.
- Hagan suyos los problemas y se sientan responsables de su solución.
- Evalúen su propio desempeño comparándolos con sus metas y objetivos personales.
- Compartan libremente su conocimiento y experiencia.
- Busquen de forma activa mejorar su competencia, su conocimiento y su experiencia.

Principio 4: Enfoque basado en procesos

Un resultado deseado se alcanza más eficientemente cuando las actividades y los recursos relacionados se gestionan como un proceso.

Cualquier actividad que utiliza recursos para transformar entradas en salidas puede considerarse un proceso. Para que las organizaciones operen de forma eficaz, tienen que identificar y gestionar numerosos procesos interrelacionados y que interactúan. A menudo la salida de un proceso forma directamente la entrada del siguiente proceso. La identificación y gestión sistemática de los procesos empleados en la organización y en particular las interacciones entre tales procesos se conocen como *"enfoque basado en procesos"*.

Una ventaja de la utilización del enfoque basado en procesos, es el control continuo que proporciona sobre los vínculos entre los procesos individuales dentro del propio sistema de procesos.

Beneficios clave:

- Costos más bajos y períodos más cortos a través del uso eficaz de los recursos.
- Resultados mejorados, consistentes y predecibles.
- Identificación y priorización de las oportunidades de mejora.

La aplicación del principio de enfoque basado en procesos conduce a lo siguiente:

- Definir sistemáticamente las actividades necesarias para obtener un resultado deseado.

- Establecer responsabilidades claras para gestionar las actividades clave.

- Analizar y medir la capacidad de las actividades clave.

- Identificar las interfaces de las actividades clave dentro y entre las funciones de la organización.

- Evaluar los riesgos, las consecuencias y los impactos de dichas actividades sobre los clientes, los proveedores y otras partes interesadas.

Principio 5: Enfoque del sistema para la gestión

Identificar, entender y gestionar los procesos interrelacionados como un sistema, contribuye a la eficacia y eficiencia de una organización en el logro de sus objetivos.

Un enfoque para desarrollar e implementar un sistema de gestión de la calidad comprende diferentes etapas tales como:

a) determinar las necesidades y expectativas de los clientes y de otras partes interesadas,

b) establecer la política y objetivos de la calidad de la organización,

c) determinar los procesos y las responsabilidades necesarias para el logro de los objetivos de la calidad,

d) determinar y proporcionar los recursos necesarios para el logro de los objetivos de la calidad,

e) establecer los métodos para medir la eficacia y eficiencia de cada proceso,

f) aplicar estas medidas para determinar la eficacia y eficiencia de cada proceso.

Beneficios clave:

- Integración y alineación de los procesos que mejor lograrán los resultados deseados.

- Capacidad de centralizar los esfuerzos en los procesos clave.

- Proporcionar confianza a las partes interesadas respecto de la consistencia, la eficacia y la eficiencia de la organización.

La aplicación del principio de enfoque de sistema para la gestión conduce a lo siguiente:

- Estructurar un sistema para lograr los objetivos de la organización en la forma más eficaz y eficiente.

- Comprender las interdependencias entre los procesos del sistema.

- Brindar una mejor comprensión de las funciones y las responsabilidades necesarias para lograr los objetivos comunes y consecuentemente reducir las barreras de funciones cruzadas.

- Establecer metas y definir la manera en que determinadas actividades dentro de un sistema deberían operar.

- Mejorar continuamente el sistema mediante la medición y la evaluación.

Principio 6: Mejora continua

La mejora continua del desempeño global de una organización debería ser un objetivo permanente de ésta.

La mejora continua del sistema de gestión de la calidad es incrementar la probabilidad de aumentar la satisfacción de los clientes y de otras partes interesadas. Acciones como las expuestas a continuación están destinadas a la mejora:

a) Análisis y evaluación de la situación existente para identificar áreas para la mejora.

b) El establecimiento de los objetivos para la mejora.

c) La búsqueda de posibles soluciones para lograr los objetivos.

d) La evaluación de dichas soluciones y su selección.

e) La implementación de la solución seleccionada.

f) La medición, verificación, análisis y evaluación de los resultados de la implementación para determinar que se han alcanzado los objetivos.

g) La formalización de los cambios.

Los resultados se revisan, cuando es necesario, para determinar oportunidades adicionales de mejora. De esta manera, la mejora es una actividad continua. La información proveniente de los clientes y otras partes interesadas, las auditorías, y la revisión del sistema de gestión de la calidad pueden utilizarse para identificar oportunidades para la mejora.

Para asegurar el futuro de la organización y la satisfacción de las partes interesadas, la dirección debería crear una cultura que implique a las personas de manera activa en la búsqueda de oportunidades de mejora del desempeño de los procesos, las actividades y los servicios.

Beneficios clave:

- Ventajas en el desempeño mediante capacidades organizacionales mejoradas.
- Alineación de las actividades mejoradas a todos los niveles de acuerdo con un propósito estratégico de la organización.
- Flexibilidad para reaccionar rápidamente ante las oportunidades.

La aplicación del principio de mejora continua conduce a lo siguiente:

- Utilizar un enfoque consistente y amplio de la organización hacia la mejora continua del desempeño de la organización.
- Proporcionar a las personas capacitación en los métodos y las herramientas de la mejora continua.
- Hacer de la mejora continua de los productos, procesos y los sistemas el objetivo de cada individuo de la organización.
- Establecer metas para guiar y medidas para trazar la mejora continua.
- Reconocer y tomar conocimiento de las mejoras.

Principio 7: Enfoque basado en hechos para la toma de decisión

Las decisiones eficaces se basan en el análisis de los datos y la información.

Basarse en el análisis de datos obtenidos a partir de medidas e información recopilada. En este contexto, la organización debería analizar los datos de sus diferentes fuentes tanto para evaluar el desempeño frente a los planes, objetivos y otras metas definidas, como para identificar áreas de mejora incluyendo posibles beneficios para las partes interesadas.

Las decisiones basadas en hechos requieren acciones eficaces y eficientes tales como:

a) métodos de análisis válidos,

b) técnicas estadísticas apropiadas,

c) tomar decisiones y llevar a cabo acciones basadas en los resultados de análisis lógicos, en equilibrio con la experiencia y la intuición.

El análisis de datos puede ayudar en gran medida a determinar la causa de los problemas existentes o potenciales y por lo tanto proporciona una guía efectiva acerca de las acciones correctivas y preventivas necesarias para la mejora.

Los resultados del análisis pueden ser utilizados por la organización para determinar:

- las tendencias,
- la satisfacción del cliente,
- el nivel de satisfacción de las otras partes interesadas,
- la eficacia y eficiencia de sus procesos,
- la contribución de los proveedores,
- el éxito de sus objetivos de mejora del desempeño,
- la economía de la calidad y el desempeño financiero y el relacionado con el entorno,
- la competitividad.

Beneficios clave:

- Decisiones informadas.
- Aumento de la capacidad para demostrar la eficacia de las decisiones anteriores mediante la referencia a los registros de los hechos.

La aplicación del principio de enfoque basado en hechos para la toma de decisión conduce a lo siguiente:

- Asegurar que los datos y la información son suficientemente exactos y confiables.

- Hacer que los datos sean accesibles para quienes los necesiten.

- Analizar los datos y la información empleando métodos válidos.

- Tomar decisiones y acciones basadas en el análisis de los hechos, equilibradas con la experiencia y la intuición.

Principio 8: Relaciones mutuamente beneficiosas con el proveedor

Una organización y sus proveedores son interdependientes, y una relación mutuamente beneficiosa aumenta la capacidad de ambos para crear valor.

Establecer relaciones con los proveedores y los aliados de la organización para promover y facilitar la comunicación con el objetivo de mejorar mutuamente la eficacia y eficiencia de los procesos que crean valor.

Existen varias oportunidades para que una organización incremente el valor a través del trabajo con sus proveedores y aliados tales como:

a) optimizando el número de proveedores y de aliados,

b) estableciendo comunicación en ambos sentidos en los niveles apropiados en ambas organizaciones para facilitar la solución rápida de problemas y evitar retrasos y disputas costosos,

c) cooperando con proveedores en la validación de la capacidad de sus procesos,

d) dando seguimiento a la habilidad de los proveedores para entregar productos conformes con el objetivo de eliminar verificaciones redundantes,

e) alentando a los proveedores a implementar programas de mejora continua del desempeño y a participar en otras iniciativas conjuntas de mejora,

f) evaluando, reconociendo y recompensando los esfuerzos y los logros de los proveedores y de los aliados.

Beneficios clave:

- Aumento de la capacidad para crear valor para ambas partes.

- Flexibilidad y velocidad de las respuestas conjuntas ante cambios del mercado o de las necesidades y expectativas de los clientes.

- Optimización de los costos y los recursos.

La aplicación del principio de relaciones mutuamente beneficiosas con el proveedor conduce a lo siguiente:

- Establecer relaciones que equilibran las ganancias a corto plazo con las consideraciones a largo plazo.

- Formación de equipos expertos y de recursos con los socios.

- Identificación y selección de los proveedores.

- Comunicación clara y abierta.

- Información y planes futuros compartidos.

- Establecer actividades conjuntas de desarrollo y mejora.

- Inspirar, alentar y reconocer las mejoras y los logros de los proveedores.

Historia de la calidad

El término de calidad no aparece hasta muy avanzada la historia de la economía y la tecnología, pero ya desde los tiempos de los jefes tribales, reyes y faraones han existido los argumentos y parámetros sobre calidad. El Código de Hammurabi (alrededor del 2150 a.C.), declaraba: *"Si un albañil construye una casa para un hombre, y su trabajo no es fuerte y la casa se derrumba matando a su dueño, el albañil será condenado a muerte"*. Los inspectores fenicios cortaban la mano a quien hacía un producto defectuoso, aceptaban o rechazaban los productos y ponían en vigor las especificaciones gubernamentales. Alrededor del año 1450 a.C., los inspectores egipcios comprobaban las medidas de los bloques de piedra con un pedazo de cordel.

Orígenes y tendencias de la calidad

En los principios de la humanidad, la sociedad habitaba en cuevas. Su principal actividad era abastecerse de comida, y vivían prácticamente de la recolección de los productos que encontraban en la naturaleza. Eran seres nómadas, y su organización era simple, y muy parecida a la de ciertos grupos de animales, tenían un líder, y todos realizaban las mismas actividades como la pesca, la caza, la recolección, etc.

Cada grupo elaboraba sus propios productos según sus necesidades, por lo que no había ningún control sobre el proceso productivo que asegurase la calidad, ya que no es lo mismo trabajar para uno mismo que producir para otra persona que tiene una opinión acerca del resultado del trabajo.

Con el paso del tiempo el hombre se dio cuenta de que podía mejorar la calidad de sus alimentos, así que decidió experimentar y mejorar desde sus armas, a sus métodos de agricultura y así fue como desarrolló su propia tecnología, de esta forma comienzan a dejar de ser seres nómadas y comienza a formarse los primeros asentamientos. En este momento, la administración de la calidad, surge como un proceso para mejorar el conocimiento y posición del hombre.

Con el crecimiento demográfico de las tribus, se fueron transformando en comunidades, y fue necesario modificar los sistemas de organización. Este mismo crecimiento exigió que las tareas se organizaran de una forma más perfeccionada, y el trabajo se especializó del tal forma que surgieron los primeros artesanos y especialistas. Es el momento en el que empieza a tener lugar una función de calidad por parte del artesano por un lado y del consumidor por otro. También se inicia el comercio, incluso a grandes distancias: el intermediario, el comerciante, hace también un papel de inspector de calidad al elegir los productos con los que va a negociar.

Época moderna

En el siglo XIII empezaron a existir los aprendices y los gremios, por lo que los artesanos se convirtieron tanto en instructores como en inspectores, ya que conocían a fondo su trabajo, sus productos y sus clientes, por lo que se empeñaban en que hubiera calidad en lo que realizaban, a este proceso se le denominó *"control de calidad del operario"*. El gobierno fijaba y proporcionaba normas y un individuo podía examinar todos los productos y establecer un patrón de calidad único. Este estado de los parámetros de aplicación de la calidad podía florecer en un mundo pequeño y local, pero con el crecimiento de la población mundial se empezó a exigir más productos, y por consecuencia, una mayor distribución a gran escala.

Así con la ayuda de la Revolución Industrial, la producción en masa de productos manufacturados se hizo posible mediante la división del trabajo y la creación de partes intercambiables; sin embargo, esto origina que el operario no tenga ni iniciativa ni libertad de acción, su trabajo viene impuesto por su puesto en el conjunto del proceso productivo. El comportamiento del obrero está condicionado por un gran número de aspectos, tales como el tipo de estructura de su empresa, la situación socio-económica de su entorno, etc., siendo para él poco significativas las necesidades del usuario final, al que no conoce, y del que no aprecia sus requerimientos. Por consiguiente, el operario no conocerá el grado de satisfacción del cliente o lo que es lo mismo, la calidad de su trabajo.

La revolución industrial

La industrialización, a finales del siglo XVIII, supuso profundos cambios sociales y económicos generados por un proceso técnico. Prácticamente desaparecen los pequeños talleres artesanales y se forman grupos de trabajadores más numerosos con atribuciones específicas o similares.

A medida que transcurre el siglo XIX, avanza la complejidad de las manufacturas, las industrias crecen y la técnica progresa. Las organizaciones consideran la necesidad de incluir un mayor número de individuos que realicen la inspección de la calidad del producto.

El sistema industrial moderno comenzó a surgir a fines del siglo XIX en los Estados Unidos, donde Frederick Taylor fue el pionero de la administración científica; suprimió la planificación del trabajo como parte de las responsabilidades de los trabajadores y capataces y las puso en mano de los ingenieros industriales, realizando estos una labor de inspección de la calidad.

La organización propuesta por Taylor de la producción se extiende rápidamente. El proceso de producción se descompone en una serie de sencillas tareas y cada operario sólo realiza una de ellas. Como resultado se obtiene una duración de formación del personal relativamente corta, se puede atender rápidamente los aumentos de producción mediante la contratación de nuevo personal y el precio de producción baja.

En el siglo XX se desarrolló una era tecnológica que permitió que las masas obtuvieran productos hasta entonces reservados sólo para las clases privilegiadas. Fue en este siglo, en el año 1918, cuando Henry Ford mediante la aplicación de técnicas de control de calidad y mejora y estandarización de procesos, introdujo en la producción de la Ford Motor Company la línea de ensamblaje en movimiento. La producción de la línea de ensamblaje dividió operaciones complejas en procedimientos sencillos, capaces de ser ejecutados por personal no especializado, dando como resultado productos de gran tecnología a bajo costo.

Pero muy pronto se hizo evidente que la prioridad del director era cumplir con los plazos establecidos para la fabricación en vez de preocuparse por la calidad. Ya que perdería su puesto si no cumplía con las demandas de producción, mientras que solo recibiría una sanción si la calidad era inferior.

Entre 1920 y 1940 la tecnología industrial cambió rápidamente. La Bell System y su subsidiaria manufacturera, la Western Electric, estuvieron a la cabeza en el control de la calidad instituyendo un departamento de ingeniería de inspección que se ocupara de los problemas creados por los defectos en sus productos y la falta de coordinación entre sus departamentos. George Edwars y Walter Shewhart, como miembros de dicho departamento, fueron sus líderes. Edwars declaró: *"Existe el control de la calidad cuando artículos comerciales sucesivos tienen sus características más cercanas al resto de sus compañeros y más aproximadamente a la intención del diseñador de lo que sería el caso si no se hiciera la aplicación. Para mí, cualquier procedimiento, estadístico u otro que obtenga los resultados que acabo de mencionar es control de calidad, cualquier otro que no obtenga estos resultados no los es"*.

En 1924, el matemático Walter Shewhart introdujo el control de calidad estadístico, lo cual proporcionó un método para controlar económicamente la calidad en medios de producción en masa. Aunque el interés primordial de Shewhart eran los métodos estadísticos, también era muy consciente de los principios de la ciencia de la administración y del comportamiento, siendo él la primera persona en hablar de los aspectos filosóficos de la calidad. El punto de vista de que la calidad tiene múltiples dimensiones es atribuible únicamente a Shewhart.

Las aportaciones realizadas por este equipo fueron muy importantes. Por primera vez en la historia, se definieron conceptos y nociones hasta entonces no entendidas, como el riesgo del proveedor, riesgo del cliente, probabilidad de aceptación o inspección media total.

En 1931 Shewart publica el libro *"Economic control of quality for manufactured products"*, este libro se convirtió en el cuaderno de bitácora de la calidad en Estados Unidos. Las empresas comienzan a utilizarlo como herramienta para conseguir sus objetivos de calidad.

En 1940 la Asociación Americana de Estándares (ASA – American Standards Association), a requerimiento del Departamento de Defensa de los Estados Unidos de América, impulsó la utilización de controles estadísticos en los productos manufacturados. Así surgieron los estándares AWS Z1.1 *"Guía del Control de Calidad"*, y AWS Z1.2 *"Método Gráfico de Control y Análisis de Datos"*. Las siglas AWS provienen de American War Standards.

Aseguramiento de la calidad

La Segunda Guerra Mundial aceleró el paso de la tecnología de la calidad. La necesidad de mejorar la calidad del producto dio por resultado un aumento en el estudio de la tecnología del control de la calidad. Fue en este medio ambiente donde se expandieron rápidamente los conceptos básicos del control de la calidad. Muchas compañías pusieron en vigor programas de certificación del vendedor. Los profesionales en el aseguramiento de la calidad desarrollaron técnicas de análisis de fallos para solucionar problemas; los técnicos de la calidad comenzaron a involucrarse en las primeras fases de diseño del producto y se iniciaron las pruebas del comportamiento ambiental de los productos.

En 1946 se instituyó la ASQC (American Society for Quality Control) y su presidente electo, George Edwars, declaró en aquella oportunidad: *"La calidad va a desempeñar un papel cada vez más importante junto a la competencia en el costo y precio de venta, y toda compañía que falle en obtener algún tipo de arreglo para asegurar el control efectivo de la calidad se verá forzada, a fin de cuentas, a verse frente a frente a una clase de competencia de la que no podrá salir triunfante"*. En ese mismo año, Kenichi Koyanagi fundó la JUSE (Union of Japanese Scientists and Engineers) con Ichiro Ishikawa como su primer presidente. Una de las primeras actividades de la JUSE fue formar el Grupo de Investigación del Control de la Calidad, cuyos miembros principales fueron Shigeru Mizuno, Kaoru Ishikawa y Tetsuichi Asaka, quienes desarrollaron y dirigieron el control de calidad japonés, incluyendo el nacimiento de los círculos de la calidad.

La calidad japonesa

Después de finalizar la Segunda Guerra Mundial Japón se encontraba al frente de la reconstrucción del país, y las fuerzas de ocupación estadounidenses decidieron apoyarlo en la reconstrucción de su economía con el fin de evitar que recuperara su capacidad bélica.

Para eso Estados Unidos envió a un grupo de expertos para ayudar en su labor. Sin embargo, antes debían ganarse la confianza de los japoneses, que los veían como meros enemigos, por lo que se lanzaban a través de la radio mensajes pro-EE.UU. Lamentablemente Japón no contaba con radios, y se propuso montar unas fábricas orientadas a su fabricación. Pero, como se contaba con mano de obra inexperta, el resultado fue la mala calidad de las radios creadas. Para sanar este problema se creó el NETL (National Electric Testing Laboratory), sin embargo poco tiempo después se reconoció que esa estrategia no era buena, y se decidió reorientar los esfuerzos a la capacitación de esta nueva generación de administradores japoneses. Esto se consiguió gracias al programa realizado por la organización llamada Unión de Científicos e Ingenieros del Japón.

Entre los temas de capacitación se incluyó el control estadístico de la calidad, este tema fue aplicado gracias a los aportes de Walter Shewhart. La JUSE vio en esta temática una razón de la victoria de los EE.UU. en la guerra, por lo que solicitaron a la CCS que les recomendaran a expertos en este tema para profundizar y reforzar el tema. Debido a que Shewhart no estaba disponible, se les recomendó un profesor de la Universidad de Columbia, que había estudiado y ampliado los temas de Shewhart; este profesor era W. Edwars Deming. Ya en 1947 Deming había estado en Japón como parte de una misión de observación económica, esto facilitó su incorporación como instructor.

En 1950 W. Edwars Deming fue invitado a hablar ante los principales hombres de negocios del Japón, quienes estaban interesados en la reconstrucción de su país al termino de la Segunda Guerra Mundial, e intentado entrar en los mercados extranjeros y cambiando la reputación del Japón de producir artículos de calidad inferior. Deming los convenció de que la calidad japonesa podría convertirse en la mejor del mundo al instituirse los métodos que el proponía.

Muchas empresas comienzan a trabajar con el concepto de Sistema Integral de Calidad, que afecta al diseño, la fabricación y la comercialización, produciéndose un fenómeno singular que afectó a la comercialización y economía industrial de muchos países, como consecuencia del despegue de la industria japonesa, aplicando los conceptos del aseguramiento de la calidad y la prevención.

Los industriales japoneses aprendieron las enseñanzas de Deming y la calidad japonesa, la productividad y su posición competitiva se mejoraron y reforzaron, para ser lo que son hoy en día. Es por ello que cada año se otorga en el Japón los Premios Deming al individuo que muestre logros excelentes en teoría o aplicación del control de la calidad por estadísticas, o aquella persona que contribuya notablemente a la difusión de las técnicas del control de calidad por estadísticas, así como a su aplicación. Algunas de las empresas japonesas que han conseguido el citado premio son Nissan, Toyota, Hitachi y Nipon Steel. En el año 1989, la Florida Power and Light Company se convirtió en la primera compañía extranjera en conseguir un premio Deming.

Calidad total

En los años sesenta y setenta, Armand V. Feigenbaum fijó los principios básicos del control de la calidad total (Total Quality Control, TQC): el control de la calidad existe en todas las áreas de negocios, desde el diseño hasta las ventas. Hasta ese momento todos los esfuerzos en la calidad habían estado dirigidos a corregir actividades, no a prevenirlas. Es así que en 1958, un equipo japonés de control de calidad, dirigido por Kaoru Ishikawa, visitó a Feigenbaum en General Electric; al equipo le gustó el nombre TQC y lo llevo al Japón; sin embargo, el TQC japonés difiere del de Feigenbaum.

Con la Guerra de Corea se incrementó aún más el énfasis en la confiabilidad y ensayos del producto final. A pesar de todos los ensayos adicionales realizados, ello no capacitaba a las firmas para hacerle frente a sus objetivos de calidad y confiabilidad, de modo que empezaron a surgir los programas del conocimiento y mejora de la calidad en las áreas de la fabricación e ingeniería.

A mediados y finales de los años sesenta los programas de la calidad se habían extendido a través de la mayoría de las grandes corporaciones estadounidenses. Esta industria ocupaba la primera posición en los mercados mundiales, mientras que Europa y Japón continuaban con su reconstrucción.

La competencia extranjera comenzó a ser una amenaza para las compañías estadounidenses en los años setenta. La calidad de los productos japoneses, en especial en las ramas automotrices y de artículos electrónicos, comenzó a sobrepasar la calidad de los productos elaborados en Estados Unidos. Los consumidores fueron haciéndose más sofisticados al decidir sus compras y empezaron a pensar en el precio y calidad en términos de la duración del producto. El aumento del interés por parte del consumidor en la calidad y competencia extranjera obligó a los administradores estadounidenses a preocuparse cada vez más por la calidad.

Mejoramiento de la calidad

El final de los años setenta y el principio de los ochenta fue marcado por un empeño en la calidad en todos los aspectos de los negocios y organizaciones de servicios, incluyendo las finanzas, ventas, personal, mantenimientos, administración, fabricación y servicio. La reducción en la productividad, los altos costos, huelgas y alto desempleo hicieron que la administración se volviera hacia el mejoramiento de la calidad como medio de supervivencia organizacional.

Hoy día muchas organizaciones se empeñan en lograr el mejoramiento de la calidad, incluyendo JUSE, ASQC, EOQC (European Organization for Quality Control). Así mismo, varios centros de estudio han establecido sus propias investigaciones para estudiar este concepto, como: las Universidades de Miami, Wisconsin, Tennessee, el Centro MIT para el Estudio de Ingeniería Avanzada y la Universidad Fordham.

La Organización Internacional de normas ISO, creada hace más de cinco décadas, desde su fundación su propósito fue mejorar la calidad, aumentar la productividad, disminuir los costos e impulsar el comercio internacional.

De este organismo surgen la familia de normas ISO 9000, que están integradas por un conjunto de modelos y documentos sobre gestión de calidad. En 1987 se publicaron las normas internacionales actuales sobre aseguramiento de la calidad. Por primera vez, cada una de ellas sirve como un modelo de calidad dirigido a determinada área de la industria, la manufactura o los servicios. En la actualidad cubren todas las funciones o posibilidades de desempeño, y tienen el objetivo de llevar la calidad o la productividad de los productos o servicios que se oferten.

La industria del software

La industria del desarrollo software es una industria joven que ha evolucionado rápidamente gracias a la aplicación de las diferentes prácticas propias del control de calidad que otras disciplinas han madurado durante décadas.

Además es evidente la influencia de la industria tradicional sobre la industria del software. La siguiente tabla, propuesta por Marciniak, pone de manifiesto la relación entre la industria tradicional y la industria del software, comparando las diferentes fases superadas por el control de calidad y su aplicación por parte de la industria general, y la industria del software.

Etapa	Descripción	Industria del Software	Industria en general
Artesanos	Se fían de la creatividad y del buen trabajo artesanal	Años sesenta	Antes del siglo XIX
Inspección	Supervisores inspeccionan la calidad antes de la liberación del producto	Años setenta	Siglo XIX
Control estadístico del proceso	Cuantificación de la calidad del producto; técnicas de muestreo	Pocas evidencias de uso	Años treinta
Aseguramiento de la calidad	Uso de estándares en los sistemas de calidad para los procesos	Años ochenta	Años cincuenta
Conformidad con la calidad	Calidad total: se eliminan los derroches y minimizan los costes	Años noventa	Años ochenta
Calidad dirigida al cliente	Calidad total dirigida hacia el cuidado del cliente y del servicio	Pocas evidencias de uso	Años noventa
Calidad dirigida al mercado	Calidad total dirigida hacia el cliente existente así como a clientes en potencia	Pocas evidencias de uso	Algunas evidencias de uso

Tabla 1. Las etapas de la calidad

Orígenes de la calidad del software

En el año 1974 aparece por primera vez el concepto de aseguramiento de la calidad del software en su sentido más amplio, fue en una especificación militar norteamericana identificada como MIL-S-52779 (AD) denominado *"Programa de Requerimientos para el Aseguramiento de la Calidad del Software"*.

Ésta solicitaba de los contratistas diferentes exigencias relacionadas con el desarrollo de software así como su administración. Los puntos más importantes eran:

- Métodos a utilizar por el contratista para evaluar diseño y documentación.
- Aprobación interna de trabajo.
- Documentación de los estándares del gobierno norteamericano para un trabajo desarrollado bajo contrato con el mismo.
- Biblioteca sobre los procedimientos de control para código y datos relacionados.
- Revisiones y auditorías, especialmente para asegurar que el software ha seguido sucesivos estados en su desarrollo.
- Control de subcontratistas del software.
- Pruebas, incluyendo planes, análisis de las especificaciones externas que aseguran la verificación, criterios de esas pruebas, control de las pruebas, certificación de los resultados, revisión de la documentación de las pruebas.

La norma norteamericana perfiló el ámbito de la práctica del aseguramiento de la calidad del software, con excepción de los procesos de mejoramiento de la calidad. Además, no identificaba métodos, técnicas, prácticas o herramientas que los contratistas debían seguir. Sólo indicaba que debía ser tenido en cuenta por adelantado y controlado para asegurar su uso durante el desarrollo.

En 1979 IEEE emitió la norma P730 *"Planes de Aseguramiento de la Calidad del Software"*, muy parecida a la norma anteriormente nombrada, aunque hizo un mayor hincapié en la documentación y uso de revisiones formales reflejando el desarrollo de software según el modelo militar *"en cascada"*, pero omitió los procedimientos de prueba.

Ambos modelos se acercaban al concepto tradicional del control de calidad en orden a vigilar el trabajo de los programadores y analistas, pero no tenían como propósito sugerir herramientas y técnicas para prevenir los defectos, métodos para dirigir revisiones de código, técnicas de pruebas o uso de datos para mejorar la calidad de futuros desarrollos.

A mediados de la década de los setenta, la fiabilidad atrajo la atención de forma importante sobre conceptos como productividad del programador o control del proyecto. Sin embargo, el aseguramiento de la calidad del software no jugaba ningún papel en este periodo.

A principios de los ochenta aparecieron las primeras actividades relacionadas con el aseguramiento de la calidad en el software, éstas se centraban en establecer estándares para la detección de errores, control de cambios, documentación y control de proyectos. En 1982, Dunn y Ullman publicaron el libro *"Quality Assurance for Computer Software"* que supuso la extensión de los conceptos relacionados con el aseguramiento de la calidad del software más allá del control del proyecto, hacia la medida del software, mejora de la calidad y calidad inherente.

A mediados de los ochenta, el concepto asociado al aseguramiento de la calidad se asoció a tres significados diferentes:

- Aproximación comprensiva a la mejora del software, proceso de programación y control del proyecto software.

- Pruebas.

- Verificación y validación.

El primer concepto, a lo largo de los ochenta, incrementó su énfasis del análisis de los atributos de la estructura del software así como en la acumulación y análisis de los datos asociados a los errores. Ya que la permanencia de los errores en el proyecto software incrementa dramáticamente su presencia a medida que el error permanece más tiempo en dicho proceso.

La normalización en la ingeniería del software

La normalización consiste en un proceso donde se elaboran guías, normas y convenciones sobre una determinada materia, con el objetivo de definir, simplificar y especificar las actividades relacionadas con la materia que se trate.

La ingeniería del Software se ha ido desarrollando en las dos últimas décadas, a través de la creación e implantación en la industria software de métodos, procedimientos, técnicas y útiles que tratan de cubrir las necesidades de cada una de las etapas del ciclo de vida de un producto software, desde la definición de sus requisitos hasta su mantenimiento una vez el producto haya sido implantado.

La creación e implantación de normas de desarrollo del software, con el objetivo de maximizar la comunicación entre los profesionales del software a través de la definición de documentos generales que se han de producir, es un desafío que tiene la ingeniería del software como medio de comunicación para transferir sus métodos, técnicas y procedimientos.

A medida que ha ido aumentando la necesidad de un software más fiable, se ha reconocido que las normas de ingeniería del software son una contribución fundamental para asegurar la producción de software de calidad.

Estándares IEEE

IEEE 730-2002	Planes de aseguramiento de la calidad del software
IEEE 829-2008	Documentación de pruebas del software
IEEE 982.1, 982.2	Diccionario estándar de medidas para producir software fiable
IEEE 1008-1993	Pruebas de unidad del software
IEEE 1012-2004	Verificación y validación del software
IEEE 1028-2008	Revisiones del software
IEEE 1044-1993	Clasificación estándar para anomalías del software
IEEE 1061-1998	Estándar para una metodología de métricas de calidad del software
IEEE 1228-1994	Planes de seguridad del software

Tabla 2. Estándares IEEE

Modelos de calidad de software

En la actualidad existen diferentes modelos de uso habitual en las empresas de software que han influenciado en el proceso de implantación de la calidad en el desarrollo de productos software.

El Instituto de Ingeniería del Software (SEI), ideó un marco de referencia para el proceso de creación de software como respuesta al requerimiento del gobierno norteamericano para la obtención de un método que permitiera valorar la capacidad de los contratistas de aplicaciones informáticas que accedían a sus licitaciones. Después de cuatro años de trabajo se presentó un modelo apoyado en el concepto de madurez al que se denominó CMM.

La respuesta europea se materializó en el denominado modelo Bootstrap, que se encuentra alineada con la norma ISO 9000. Esta norma, a su vez, propuso la norma ISO 90003, guía de aplicación de la norma ISO 9001 para compañías software (ambas normas serán estudiadas posteriormente).

Estas iniciativas (CMM, Bootstrap...) dieron lugar, al inicio de la década de los noventa, al sentimiento generalizado de la necesidad de promover un estándar internacional que armonizara los modelos de referencia existentes.

El proyecto SPICE, promovido por ISO, surgió como un esfuerzo de colaboración internacional que debía materializarse en un nuevo estándar para la valoración del proceso de software. El grupo de trabajo que llevaría a cabo esta labor (WG10) contaría con un equipo central de profesionales con dedicación exclusiva, además de aportaciones de investigadores, estudiosos y profesionales de más de veinte países.

SPICE debía proporcionar el soporte necesario para la elaboración de un nuevo estándar. La realización de pruebas de campo sería una labor fundamental de la que se deberían extraer los datos oportunos y derivar los análisis que posibilitarían la mejora del modelo en sus diferentes borradores.

Entre 1993 y 1995 el borrador del producto fue desarrollado y se realizaron las primeras pruebas de campo. En 1996 se reflejaron diferentes cambios en la norma para ajustarla a los datos recogidos en las pruebas efectuadas. En octubre de ese mismo año se celebró un encuentro en Méjico del WG10 en el que la comunidad internacional, por primera vez, pudo conocer las partes que definen el modelo. En la actualidad el proyecto ha alcanzado el llamado Informe Técnico.

Hoy día, SPICE sigue conociéndose con este nombre, aunque también se conoce con el nombre de ISO 15504.

Capítulo 2: Aseguramiento de la calidad

Introducción

Como se citó en el primer apartado, en la actualidad las organizaciones o empresas de desarrollo software están implantando diferentes modelos o normas con las que conseguir establecer el aseguramiento de la calidad en todo el desarrollo de un producto software.

En esta sección, se va a llevar a cabo un estudio de los apartados más influyentes de diferentes normativas o estándares que se deben aplicar a la hora de realizar el desarrollo de un producto software. A pesar de la gran diversidad de normativas que se pueden encontrar, se han seleccionado las normas ISO 9001:2000 e ISO 90003:2005, para el aseguramiento de la calidad, dada la gran importancia que tienen ambas normas y el nivel de aceptación e implantación en el sector.

Cuando se desarrollo este capítulo, la norma ISO 9001:2008 todavía no había sido publicada. Aún así, posteriormente, se revisaron las diferencias con la norma ISO 9001:2000 y viendo que las diferencias no son muy grandes en cuento a los apartados expuestos, finalmente se decidió continuar con el estudio de la norma ISO 9001:2000, ya que la norma ISO 90003:2005, es una guía de aplicación de la norma ISO 9001:2000.

Norma ISO 9001:2000

Introducción

En todo desarrollo de proyecto software el cliente da por hecho que el producto que está adquiriendo es de calidad. Cumpliendo todos aquellos requisitos y funcionalidades que expuso en el proceso de toma de requisitos.

El objetivo de esta norma es especificar los requisitos para un sistema de gestión de la calidad, cuando una organización necesita demostrar su capacidad para proporcionar de forma coherente productos que satisfagan los requisitos del cliente y aspire a mejorar esta satisfacción a través de la aplicación eficaz del sistema, incluyendo los procesos de mejora continua del sistema y el aseguramiento de la conformidad con los requisitos del cliente.

Para ello promueve la adopción por parte de las empresas a que enfoquen sus sistemas de gestión de la calidad en procesos. La identificación y gestión de estos puede hacer que el sistema de gestión de la calidad sea más eficaz a la hora de satisfacer los requisitos del cliente. Con este enfoque se consigue un control continuo del proyecto y se enfatiza la importancia de la comprensión y cumplimiento de los requisitos, la necesidad de considerar los procesos en términos que aporten valor, la obtención de resultados del desempeño y eficacia del proceso y la mejora continua de los procesos con base en mediciones objetivas.

A continuación, dentro de todos los procesos que componen esta norma, se muestran aquellos puntos de la norma que se centran más en el desarrollo de software, con el objetivo de enmarcar aquellos procesos que resultan un factor clave a la hora de desarrollar y mantener un producto software.

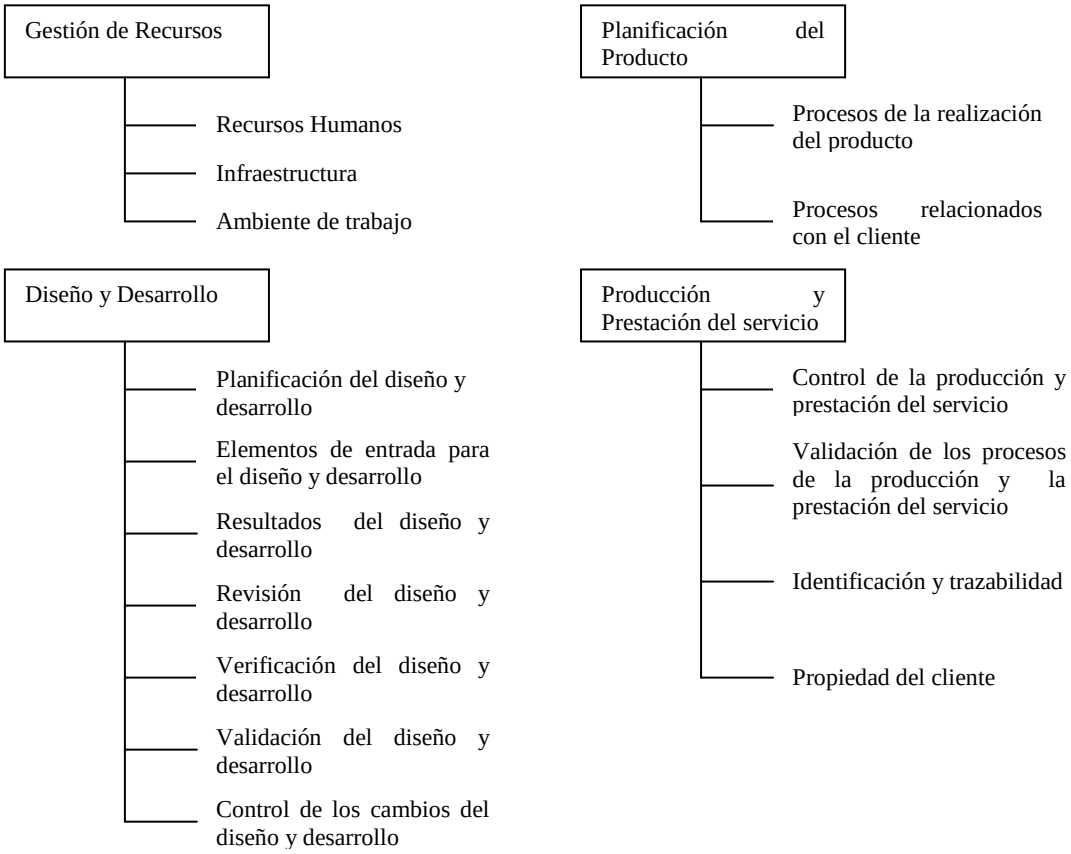

Imagen 2. Procesos clave Norma ISO 9001

Gestión de recursos

Recursos humanos

El personal que realice trabajos que afecten a la calidad del producto debe ser competente con base en la educación, formación, habilidades y experiencia apropiadas.

El requisito que se define en este punto, es que el personal sea competente, pero para ello antes la organización necesita identificar los requisitos de cada uno de los puestos. Este apartado incluye al personal implicado en la alta dirección, la gestión de los recursos, la realización del producto, y los procesos de medición, análisis y mejora. Por ello la organización deberá identificar los cursos de formación, seminarios, la formación en el puesto de trabajo o cualquier otro tipo de formación para que todos los empleados involucrados en el sistema de gestión de la calidad sean competentes. Evaluar la eficacia de las acciones tomadas y asegurarse de que su personal es consciente de la pertinencia e importancia de sus actividades y de cómo contribuyen al logro de los objetivos de la calidad.

Infraestructura

La organización debe determinar, proporcionar y mantener la infraestructura necesaria para lograr la conformidad de los requisitos del producto. La infraestructura incluye, cuando sea aplicable:

- *Edificios, espacio de trabajo y servicios asociados.*
- *Equipo para los procesos (tanto hardware como software).*
- *Servicios de apoyo (tales como transporte y comunicación).*

Dentro de este punto, en el desarrollo de software dentro de un equipo de trabajo lo más importante es disponer de equipos para los procesos y servicios de apoyos. Los recursos físicos proporcionan la base para la realización del trabajo necesario. El diseño y el desarrollo de software requieren salas de trabajo adecuadas, herramientas de programación y procesadores adecuados.

Ambiente de trabajo

La organización debe determinar y gestionar el ambiente de trabajo necesario para lograr la conformidad con los requisitos del producto.

Esto se puede considerar como una combinación de factores humanos y físicos. Algunos factores humanos a tener en cuenta porque puede afectar al desarrollo de un producto son:

- Métodos de trabajo.
  Manera en que la organización lleva a cabo su trabajo. Siempre es aconsejable tener un método de trabajo motivador y dinámico para que el personal se encuentre a gusto y rinda mejor.

- Normas y consejos de seguridad.

- Ergonomía.

Algunos factores físicos que afectan al ambiente de trabajo pueden ser los siguientes:

- Calor	- Higiene	- Vibración
- Ruido	- Humedad	- Contaminación
- Luz	- Limpieza	

Estos factores son aconsejables que se lleven a cabo en grupos de desarrollo software, al igual que en cualquier otra organización que desempeñe otra función, ya que dependiendo de estos un proyecto puede desarrollarse de una forma más amena. Además de asegurar el buen funcionamiento de los equipos informáticos.

Planificación del producto

Planificación de la realización del producto

La organización debe planificar y desarrollar los procesos necesarios para la realización del producto. La planificación de la realización del producto debe ser coherente con los requisitos de los otros procesos del sistema de gestión de la calidad.

Durante la planificación de la realización del producto, la organización debe determinar, cuando sea apropiado, lo siguiente:

- *los objetivos de la calidad y los requisitos para el producto;*
- *la necesidad de establecer procesos, documentos y de proporcionar recursos específicos para el producto;*
- *las actividades requeridas de verificación, validación, seguimiento, inspección y ensayo/prueba específicas para el producto, así como los criterios para la aceptación del mismo;*
- *los registros que sean necesarios para proporcionar evidencia de que los procesos de realización y el producto resultante cumplen los requisitos.*

En organizaciones de desarrollo software, es vital tener constancia de hasta que punto del proyecto se ha llegado a implementar. El uso de diagramas de flujo puede resultar apropiado para garantizar que se abordan todos los pasos del proceso en lo que respecta a disponibilidad de documentación, consecución de los objetivos marcados para ese día/semana…

Procesos relacionados con el cliente

- Determinación de los requisitos relacionados con el producto

La organización debe determinar:

- *Los requisitos especificados por el cliente, incluyendo los requisitos para las actividades de entrega y las posteriores a la misma.*

- *Los requisitos no establecidos por el cliente pero necesarios para el uso especificado o para el uso previsto.*

- *Los requisitos legales y reglamentarios relacionados con el producto.*

Para la realización de un proyecto software, la complejidad que conlleva la ejecución de este punto y lo importante que resulta para llevar a cabo un proyecto de forma que el cliente se sienta satisfecho, requiere la atención de todas las partes de la organización como la cuestión más importante del proyecto, ya que de ello depende la comprensión de lo que se debe realizar en el proyecto y con ello lograr la satisfacción del cliente. Dada la importancia de lograr una clara comprensión de los requisitos del cliente se deberán tener cuantas reuniones sean necesarias hasta que los requisitos del sistema queden totalmente claros.

- Revisión de los requisitos relacionados con el producto

La organización debe revisar los requisitos relacionados con el producto. Esta revisión debe efectuarse antes de que la organización se comprometa a proporcionar un producto al cliente y debe asegurarse de que:

- *Están definidos los requisitos del producto.*
- *Están resueltas las diferencias existentes entre los requisitos del contrato o pedido y los expresados previamente.*
- *La organización tiene la capacidad para cumplir con los requisitos definidos.*

Cuando el cliente no proporcione una declaración documentada de los requisitos, la organización debe confirmar los requisitos del cliente antes de la aceptación.

Al igual que el apartado anterior es de aplicación a todo tipo de productos software. No es nada recomendable llegar a un acuerdo con el cliente sin antes haber revisado los requisitos del sistema y tener una declaración documentada de los requisitos confirmada por parte del cliente.

- Comunicación con el cliente

La organización debe determinar e implementar disposiciones eficaces para la comunicación con los clientes, relativa a:

- *La información sobre el producto.*
- *Las consultas, contratos o atención de pedidos, incluyendo las modificaciones.*
- *La retroalimentación del cliente, incluyendo sus quejas.*

Los acuerdos identificados y puestos en práctica deberían resultar adecuados para la organización en lo que respecta a sus productos. Las organizaciones deberían disponer de un proceso que garantizase la existencia de unas comunicaciones adecuadas con los clientes en lo que respecta a la información del producto, las consultas sobre el mismo y posibles modificaciones.

En un proceso de desarrollo software se hace vital tener un buen proceso de relación con el cliente, ya que en cualquier momento se puede producir la modificación de algunos de los requisitos lo que podría conllevar a realizar una gran modificación en el proyecto. Así, mediante un buen proceso de relación, cualquier evento que pueda influir en el proyecto será detectado en el instante preciso.

Diseño y desarrollo

Planificación del diseño y desarrollo

La organización debe planificar y controlar el diseño y el desarrollo del producto.

Durante la planificación del diseño y desarrollo, la organización debe determinar:

- *Las etapas del diseño y desarrollo.*
- *La revisión, verificación y validación apropiadas para cada etapa del diseño y desarrollo.*
- *Las responsabilidades y autoridades para el diseño y desarrollo.*

La organización debe gestionar las interfaces entre los diferentes grupos involucrados en el diseño y desarrollo para asegurarse de una comunicación eficaz y una clara asignación de responsabilidades.

El propósito de este apartado es garantizar que la organización planifica y controla las etapas de diseño y desarrollo del proyecto. Con ello se intenta maximizar las probabilidades de que el proyecto satisfaga los requisitos definidos.

La planificación debe presentar el nivel de detalle necesario para alcanzar los objetivos del diseño y desarrollo, evitando generar una excesiva cantidad de documentos. Un enfoque típico consiste en crear algún tipo de diagrama de flujo del proyecto que incorpore la información pertinente sobre personal, calendario e interrelaciones. Por ejemplo, los gráficos de Gantt o PERT. Es necesario determinar las etapas del proyecto y definir las responsabilidades, la autoridad y las interfaces. Igualmente, es preciso establecer los requisitos para la incorporación de la revisión, la verificación y la validación al proyecto de diseño y desarrollo.

Elementos de entrada para el diseño y desarrollo

Deben determinarse los elementos de entrada relacionados con los requisitos del producto y mantenerse registros. Estos elementos de entrada deben incluir:

- *Los requisitos funcionales y de desempeño.*
- *Los requisitos legales y reglamentarios aplicables.*
- *La información proveniente de diseños previos similares, cuando sea aplicable.*
- *Cualquier otro requisito esencial para el diseño y desarrollo.*

Estos elementos deben revisarse para verificar su adecuación. Los requisitos deben estar completos, sin ambigüedades y no deben ser contradictorios.

El propósito de este apartado es garantizar el desarrollo y la documentación de una especificación de requisitos, incluyendo la idoneidad del producto para satisfacer las necesidades del cliente.

La fase de desarrollo no debería comenzar hasta que exista un documento en el que se haya determinado todos los requisitos de las numerosas áreas que hay que tener en cuenta, como pueden ser las normas nacionales, las normas de la organización, las necesidades y expectativas del cliente, los costes... Y este documento haya sido aceptado por las diferentes partes.

Resultados del diseño y desarrollo

Los resultados del diseño y desarrollo deben proporcionarse de tal manera que permitan la verificación respecto a los elementos de entrada para el diseño y desarrollo y deben aprobarse antes de su liberación.

Los resultados de diseño y desarrollo deben:

- *Cumplir los requisitos de los elementos de entrada para el diseño y desarrollo.*
- *Proporcionar información apropiada para la compra, la producción y la prestación del servicio.*
- *Contener o hacer referencia a los criterios de aceptación del producto.*
- *Especificar las características del producto que son esenciales para el uso seguro y correcto.*

Los resultados del diseño y el desarrollo deben proporcionarse de forma que puedan ser utilizados para la posterior verificación. Esto significa que debe haber evidencia objetiva de que el diseño y el desarrollo se han llevado a cabo de conformidad con los requisitos definidos al comienzo del proyecto.

La documentación de los resultados de un proyecto de diseño y desarrollo demuestra que el producto hará lo que se espera que haga. En un producto software es muy importante garantizar que no interferirá en el funcionamiento de otros productos software.

La norma además, exige que se proporcione información para facilitar la realización del producto. En el caso del software, generalmente no es necesario abordar el punto: *"Proporcionar información apropiada para la compra, la producción y la prestación del servicio"*. El siguiente punto requiere una inequívoca declaración de los requisitos que debe satisfacer el producto con el objeto de que sea aceptable para los clientes.

Se espera que los resultados de este proceso incluyan cualquier información relacionada con la producción o usos seguros y correctos del producto. Por último, estos resultados deben ser aprobados antes de la liberación del producto.

Revisión del diseño y desarrollo

En las etapas adecuadas, deben realizarse revisiones sistemáticas del diseño y desarrollo de acuerdo con lo planificado:

- *Evaluar la capacidad de los resultados de diseño y desarrollo para cumplir los requisitos.*

- *Identificar cualquier problema y proponer las acciones necesarias.*

Los participantes de dichas revisiones deben incluir representantes de las funciones relacionadas con la(s) etapa(s) de diseño y desarrollo que está(n) revisando. Deben mantenerse registros de los resultados de las revisiones y de cualquier acción necesaria.

La revisión del diseño y el desarrollo se utiliza para garantizar la oportuna liberación de un nuevo producto que satisface por completo las necesidades del cliente. Igualmente, tiene como finalidad tratar otros asuntos aparte de la cuestión de si el producto cumplirá los requisitos específicos. Su objetivo es ocuparse de las capacidades asociadas a un nuevo producto: capacidad de fabricación, suministro, comprobación, inspección, servicio de mantenimiento... El propósito de las revisiones del diseño y el desarrollo es identificar problemas, discutir posibles soluciones y determinar el seguimiento apropiado.

Debe mantenerse registros de las revisiones del diseño y el desarrollo. Como mínimo debería incluir registros de los problemas y las acciones propuestas.

Es un elemento fundamental del proceso de diseño y desarrollo software. Cuando en los proyectos de desarrollo software se realizan exhaustivas revisiones del diseño y el desarrollo, incluyendo revisiones del diseño y el desarrollo de los planes de comprobación del software, lo habitual es que se reduzcan los ciclos de desarrollo y que disminuyan los costes de operación y mantenimiento.

Verificación del diseño y desarrollo

Se debe realizar la verificación de acuerdo con lo planificado para asegurarse de que los resultados del diseño y desarrollo cumplen los requisitos de los elementos de entrada del diseño y desarrollo. Deben mantenerse registros de los resultados de la verificación y de cualquier acción que sea necesaria.

La verificación es un paso que puede aparecer en distintas etapas del proceso de diseño y desarrollo. Analiza el producto una vez que los encargados del desarrollo han concluido su trabajo para garantizar que el resultado satisface los requisitos especificados. Esta se puede llevar a cabo mediante la revisión y el análisis de los resultados de los ensayos, efectuando cálculos alternativos, realizando comprobaciones adicionales del producto o de sus componentes.

En caso de que surja algún problema durante esta etapa, debe documentarse e identificarse las acciones de seguimiento. Estas acciones pueden requerir una nueva comprobación de los resultados obtenidos cotejándolos con las especificaciones y los requisitos de los elementos de entrada y una revalidación del producto con anterioridad a su liberación.

Validación del diseño y desarrollo

Se debe realizar la validación del diseño y desarrollo de acuerdo con lo planificado para asegurarse de que el producto resultante es capaz de satisfacer los requisitos para su aplicación especificada o uso previsto, cuando sea conocido. Siempre que sea factible, la validación debe completarse antes de la entrega o implementación del producto. Deben mantenerse registros de los resultados de la validación y de cualquier acción que sea necesaria.

El objetivo de esta etapa es garantizar que el resultado del diseño y el desarrollo, satisface las necesidades definidas del usuario. Por norma general, la validación se lleva a cabo tras la exitosa verificación del diseño y el desarrollo.

Aclaración: Diferencia existente entre verificación y validación. La verificación se ocupa de la conformidad de los requisitos, mientras que la validación se ocupa de la satisfacción de las necesidades definidas del usuario.

Por ejemplo, si el resultado de un proyecto de diseño de un producto software funcionase de acuerdo con lo establecido en la especificación de requisitos definidos por el usuario, pero hiciera que cualquier otro tipo de producto software dejase de funcionar o fallase, este producto cumpliría el requisito del apartado de verificación, pero no el del apartado de validación.

Control de cambios del diseño y desarrollo

Los cambios del diseño y desarrollo deben identificarse y deben mantener los requisitos. Los cambios deben revisarse, verificarse y validarse, según sea apropiado, y aprobarse antes de su implementación. La revisión de los cambios del diseño y desarrollo debe incluir la evaluación del efecto de los cambios en las partes constitutivas y en el producto ya entregado.

Deben mantenerse registros de los resultados de la revisión de los cambios y de cualquier acción que sea necesaria.

Durante las tareas de diseño y desarrollo de un proyecto, suelen producirse cambios en los requisitos definidos en la fase inicial. Dichos cambios se pueden producir por múltiples motivos, como pueden ser entre otros:

- Omisiones que se ponen de manifiesto una vez ha dado comienzo la labor de diseño y desarrollo.
- Cambios solicitados por el cliente.
- Problemas surgidos durante la revisión del diseño y el desarrollo.
- Problemas surgidos durante el proceso de verificación o validación.

Cualquier tipo de cambio que se produzca en el diseño de un producto, ya sea durante el proceso de diseño y desarrollo, durante la producción o tras el suministro del producto al cliente, debe identificar y deben mantenerse registros. En especial en el desarrollo de proyectos software, donde el control de la configuración es una cuestión fundamental.

Los cambios que se produzcan deberían probarse por medio de los procesos de revisión, verificación y validación del diseño y el desarrollo, para evitar efectos negativos en otros elementos del producto.

Los productos software al ser un bien intangible necesitan llevar a cabo la consecución de todos estos apartados. Ya que si no se llevase un control de una forma tan exhaustiva de cada uno de estos apartados, llegar a los objetivos marcados al inicio del proyecto sería mucho más difícil, lo que llevaría un consumo de tiempo innecesario y perdida de dinero, además crearía el descontento del cliente lo que podría incurrir en la pérdida del mismo.

Producción y prestación del servicio

Control de la producción y prestación del servicio

La organización debe planificar y llevar a cabo la producción y la prestación del servicio bajo condiciones controladas. Las condiciones controladas deben incluir, cuando sea aplicable:

- *La disponibilidad de información que describa las características del producto.*
- *La disponibilidad de instrucciones de trabajo, cuando sea necesario.*
- *El uso del equipo apropiado.*
- *La disponibilidad y uso de dispositivos de seguimiento y medición.*
- *La implementación del seguimiento y de la medición.*
- *La implementación de actividades de liberación, entrega y posteriores a la entrega.*

La organización debe planificar y llevar a cabo la producción y la prestación del servicio bajo condiciones controladas. Por ello, la organización debería controlar las operaciones de procesos teniendo en cuenta varios factores. Se debe determinar los procesos de producción y prestación del servicio que necesitan ser controlados y los resultados que deben obtenerse en cada etapa del procesamiento.

Es preciso analizar los equipos específicos que se necesitan para lograr las especificaciones del producto.

La organización necesita determinar los criterios de aceptabilidad de estos procesos y ha de realizar evaluaciones conforme a estos criterios. Entre estas evaluaciones se pueden realizar controles o comprobaciones del resultado del proceso, las características del mismo…

Centrándose en la creación de productos software, es esencial controlar las etapas finales de desarrollo hasta llegar a la reproducción del código y los posteriores procesos de instalación y servicio posventa o mantenimiento.

Entre los controles del proceso deberían incluirse procedimientos documentados desde el comienzo de la fase de diseño y debería prolongarse a lo largo de todo el ciclo de vida del producto software. Así se contribuye al control del diseño, el desarrollo, suministro y el posterior uso del software.

Validación de los procesos de producción y de la prestación del servicio

La organización debe validar aquellos procesos de producción y de prestación del servicio donde los productos resultantes no puedan verificarse mediante actividades de seguimiento o medición posteriores. Esto incluye a cualquier proceso en el que las deficiencias se hagan aparentes únicamente después de que el producto esté siendo utilizado o se haya prestado el servicio.

La validación debe demostrar la capacidad de estos procesos para alcanzar los resultados planificados.

La organización debe establecer las disposiciones para estos procesos, incluyendo, cuando sea aplicable:

- *Los criterios definidos para la revisión y aprobación de los procesos.*
- *La aprobación de equipos y calificación del personal.*
- *El uso de métodos y procedimientos específicos.*
- *Los requisitos de los registros y la revalidación.*

El software no se puede verificar por completo mediante ensayos. Todo software necesita ser creado por medio de procesos controlados siguiendo un modelo (en este caso al estar realizando el estudio de la norma 9001 sería siguiendo el modelo de diseño y desarrollo expuesto anteriormente). Los métodos y el alcance de la validación de los procesos dependen considerablemente en función de lo crítico y el uso que vaya a recibir el software. Para poder garantizar que el software cumple los requisitos especificados, las cualidades del personal, los equipos y las metodologías de desarrollo software son aspectos muy importantes a tener en cuenta.

Identificación y trazabilidad

Cuando sea apropiado, la organización debe identificar el producto por medios adecuados, a través de toda la realización del producto.

La organización debe identificar el estado del producto con respecto a los requisitos de seguimiento y medición.

Cuando la trazabilidad sea un requisito, la organización debe controlar y registrar la identificación única del producto.

La identificación y la trazabilidad son aspectos independientes, pero relacionados. Debe determinarse el grado necesario de identificación de los productos, incluyendo cualquier requisito de localización de componentes y materiales que estén relacionados de forma única con el producto. Una parte esencial de la identificación del producto es su estado en lo que respecta a la satisfacción de los requisitos en las diferentes etapas de la producción, el almacenamiento y la entrega.

La organización debe identificar el producto y sus componentes con el debido detalle. Así pues, la trazabilidad está estrechamente relacionada con la identificación. Deberían definirse los registros necesarios para garantizar la trazabilidad.

La gestión de la configuración del software requiere que cada una de las versiones de un elemento de configuración se identifique por medios apropiados. Igualmente, es preciso mantener un registro del estado de las fases de la verificación y los ensayos que se han completado. También deben mantenerse los resultados obtenidos por el producto o los componentes del producto en cada etapa del ciclo de desarrollo.

Propiedad del cliente

La organización debe cuidar los bienes que son propiedad del cliente mientras estén bajo el control de la organización o estén siendo utilizados por la misma. La organización debe identificar, verificar, proteger y salvaguardar los bienes que son propiedad del cliente suministrados para su utilización o incorporación dentro del producto. Cualquier bien que sea propiedad del cliente que se pierda, deteriore o que de algún otro se considere inadecuado para su uso debe ser registrado y comunicado al cliente.

La propiedad del cliente es aquel producto que pertenece al cliente y es suministrado a la organización para su uso a la hora de satisfacer los requisitos del acuerdo entre ambas partes. La organización se compromete a proteger el producto mientras esté en su poder.

Referente al software, este apartado puede ser un factor importante en aquellas operaciones relacionadas con este. Por ejemplo un cliente proporciona un código fuente a un contratista de programación para que lleve a cabo su modificación y así incorporar más funcionalidad. La organización debe tomar precauciones para proteger la funcionalidad original del software. Existen acuerdos detallados que definen estas relaciones.

Norma ISO 9003:2005

Introducción

Con el objetivo de especificar de una forma más amplia los puntos que anteriormente se han expuesto, se ha decidido introducir también esta norma que proporciona las directrices, o pasos a seguir, para las organizaciones en la aplicación de la normativa ISO 9001:2000 en la compra, provisión, desarrollo, operación y mantenimiento de software.

Identifica los temas que tendrían que ser tratados y es independiente a la tecnología, modelos de ciclo de vida, procedimientos de desarrollo…

Tanto en esta norma como en la ISO-9001:2000, la palabra *debería* se utiliza para expresar una recomendación entre posibilidades y *puede* para indicar una opción de acción permisible dentro de los límites de esta norma internacional.

Las referencias que se hagan a la norma ISO-9001:2000 se mostrarán en letra cursiva, para que resulte más fácil su identificación.

Gestión de recursos

Recursos humanos

Generalidades: *El personal que realice trabajos que afecten a la calidad del producto debe ser competente con base en la educación, formación, habilidades y experiencia apropiadas.*

Competencia, toma de conciencia y formación: *La organización debe:*

a) *determinar la competencia necesaria para el personal que realice trabajos que afecten a la calidad del producto.*

b) *proporcionar formación u otras necesidades para satisfacer dichas necesidades.*

c) *evaluar la eficacia de las acciones tomadas.*

d) *asegurarse de que su personal es consciente de la pertinencia e importancia de sus actividades y de cómo contribuyen al logro de los objetivos.*

e) *mantener los registros adecuados de la educación, formación, habilidades y experiencia.*

Se deberían determinar las necesidades de formación considerando requisitos de notación, métodos de diseño, lenguajes específicos de programación, herramientas, técnicas y recursos de ordenador a usar en el desarrollo y en la gestión del producto/proyecto software.

Las tecnologías empleadas en el desarrollo, operación y mantenimiento de software deberían ser monitorizadas y evaluadas en orden a determinar los requisitos para actualizar los perfiles del personal.

Infraestructura

La organización debe determinar, proporcionar y mantener la infraestructura necesaria para lograr la conformidad de los requisitos del producto. La infraestructura incluye, cuando sea aplicable:

a) *Edificios, espacio de trabajo y servicios asociados.*

b) *Equipo para los procesos (tanto hardware como software).*

c) *Servicios de apoyo (tales como transporte y comunicación).*

La infraestructura debería incluir hardware, software, herramientas y servicios para el desarrollo, operación y mantenimiento del software.

La infraestructura puede incluir herramientas software que soporten los procesos de diseño y desarrollo, las más importantes para el desarrollo de software son:

Herramientas, tales como para análisis, diseño y desarrollo, gestión de la configuración, pruebas, gestión de proyectos, documentación; entornos de aplicaciones de desarrollo y soporte; herramientas de gestión del conocimiento, intranet, extranet; herramientas de redes, incluyendo seguridad, backup, antivirus, firewall; control de accesos; librerías de software.

Dependiendo del caso puede que estas herramientas sean apropiadas o no para el objetivo, en caso de serlo el uso de estas puede documentarse con guías apropiadas, y su uso revisado, cuando sea apropiado, para determinar si hay necesidad de mejorar o actualizarlas.

Ambiente de trabajo

Dado que este punto se encuentra bastante detallado en este mismo apartado de la norma ISO-9001:2000, para evitar redundancia innecesaria es preferible revisar este apartado en la norma ISO-9001:2000.

Realización del producto

Planificación y realización del producto

La organización debe planificar y desarrollar los procesos necesarios para la realización del producto. La planificación de la realización del producto debe ser coherente con los requisitos de los otros procesos del sistema de gestión de la calidad.

Durante la planificación de la realización del producto, la organización debe determinar, cuando sea apropiado, lo siguiente:

- *los objetivos de la calidad y los requisitos para el producto;*
- *la necesidad de establecer procesos, documentos y de proporcionar recursos específicos para el producto;*
- *las actividades requeridas de verificación, validación, seguimiento, inspección y ensayo/prueba específicas para el producto, así como los criterios para la aceptación del mismo;*
- *los registros que sean necesarios para proporcionar evidencia de que los procesos de realización y el producto resultante cumplen los requisitos.*

- Ciclo de vida software

Los procesos, actividades y tareas deberían planificarse y realizarse usando un modelo de ciclo de vida apropiado para las necesidades del producto software, considerando diferentes pautas como pueden ser el tamaño, complejidad, seguridad... La norma ISO 9001 está destinada a aplicarse sin tener en cuenta los modelos del ciclo de vida usados y no está destinada a indicar un modelo de ciclo de vida.

Por ejemplo, el diseño y desarrollo puede ser un proceso evolutivo y los procedimientos pueden por tanto necesitar cambiarse y actualizarse cuando progrese el proyecto, después de la consideración de los cambios de las actividades y tareas relacionadas.

Debería considerarse la adaptabilidad del método de diseño y desarrollo al tipo de tarea o producto y la compatibilidad de la aplicación, los métodos y las herramientas a usar.

- Planificación de la calidad

La planificación de la calidad suministra los medios para adaptar la aplicación del sistema de gestión de calidad a un proyecto o producto. La planificación de la calidad puede incluir o hacer referencia a procedimientos genéricos y/o específicos del proyecto o producto. Esta debe ser revisada a lo largo del proceso de diseño y desarrollo, y los elementos afectados con cada fase deberían ser completamente definidos cuando comience esa fase.

La planificación de la calidad es particularmente útil para clarificar los objetivos de calidad limitados para el software que está siendo diseñado para un propósito limitado.

Procesos relacionados con el cliente

- Determinación de los requisitos relacionados con el producto

La organización debe determinar:

- *Los requisitos especificados por el cliente, incluyendo los requisitos para las actividades de entrega y las posteriores a la misma.*

- *Los requisitos no establecidos por el cliente pero necesarios para el uso especificado o para el uso previsto.*

- *Los requisitos legales y reglamentarios relacionados con el producto.*

Requisitos relacionados con el cliente.

El software puede desarrollarse como parte de un contrato, como un producto disponible para un sector de mercado, como software embebido en un sistema; o apoyo de los procesos de negocio de la organización. La determinación de los requisitos es aplicable en todas estas circunstancias.

Los requisitos pueden ser proporcionados por el cliente, desarrollados por la organización o desarrollados conjuntamente.

Cuando los requisitos son proporcionados y acuerdan en la forma de una especificación de sistema, los métodos deberían estar en su lugar correspondiente para asignarlos en los elementos de hardware y software con cualesquiera especificaciones de interfaz adecuadas.

En algunas situaciones, los requisitos pueden no estar definidos en su completitud en la aceptación del contrato, tomándose algunos requisitos a lo largo del desarrollo del software.

Los requisitos deben expresarse de forma clara y concisa, evitando ambigüedades, que faciliten la validación durante la aceptación del producto.

- Revisión de los requisitos relacionados con el producto

La organización debe revisar los requisitos relacionados con el producto. Esta revisión debe efectuarse antes de que la organización se comprometa a proporcionar un producto al cliente y debe asegurarse de que:

- *Están definidos los requisitos del producto.*
- *Están resueltas las diferencias existentes entre los requisitos del contrato o pedido y los expresados previamente.*
- *La organización tiene la capacidad para cumplir con los requisitos definidos.*

Cuando el cliente no proporcione una declaración documentada de los requisitos, la organización debe confirmar los requisitos del cliente antes de la aceptación.

Como se dice en la norma ISO 9001:2000, no es recomendable llegar a un acuerdo con el cliente sin antes haber revisado los requisitos del sistema y tener una declaración documentada de los requisitos confirmada por parte del cliente.

Igualmente deben tener en cuenta los intereses de la organización y riesgos.

Los intereses de la organización, son aspectos que pueden ser relevantes durante la revisión por la organización de las ofertas de software o contratos, pueden incluir entre otros:

a) viabilidad del cumplimiento y validación de los requisitos y características del producto.

b) normas y procedimientos de diseño y desarrollo de software a utilizar.

c) requisitos de replicación y distribución.

d) aspectos relacionados con el cliente:

    - Procesos del ciclo de vida impuestos por el cliente.

    - Período obligatorio de la organización para suministrar copias y la capacidad de lectura de copias maestras.

e) aspectos de gestión, como pueden ser:

    - Gestión de riesgos.

    - Responsabilidad de la organización con respecto al trabajo subcontratado.

    - Calendario.

    - Requisitos de instalación.

Los riesgos se pueden incluir cuando se revisan los requisitos relativos al producto, como son:

- Aspectos de criticidad, seguridad de las personas y seguridad industrial.
- Capacidades y experiencia de la organización.
- Fiabilidad de las estimaciones de los recursos y de la duración requerida por cada actividad.
- Baja calidad o disponibilidad de herramientas y software suministrado.

- Comunicación con el cliente

La organización debe determinar e implementar disposiciones eficaces para la comunicación con los clientes, relativa a:

- *La información sobre el producto.*
- *Las consultas, contratos o atención de pedidos, incluyendo las modificaciones.*
- *La retroalimentación del cliente, incluyendo sus quejas.*

El método de comunicación puede variar dependiendo del tipo de acuerdo contractual, y sobre el objeto y alcance del contrato de desarrollo, operación o mantenimiento.

Las siguientes directrices de comunicación con clientes se separan en consejos para los procesos del ciclo de vida para el desarrollo y para la operación/mantenimiento.

Durante el ciclo de desarrollo, las revisiones conjuntas que involucran a la organización y al cliente deben ser planificadas con regularidad, o en hitos significativos para el proyecto para cubrir aspectos acerca de la información del producto, o investigaciones, contratos o modificaciones.

Durante las operaciones y el mantenimiento, las fuentes de información que conciernen a la comunicación con el cliente en las operaciones y el mantenimiento pueden incluir, información del producto, consultas, contratos o modificaciones, y retroalimentación del cliente.

Diseño y desarrollo

Planificación del diseño y desarrollo

La organización debe planificar y controlar el diseño y el desarrollo del producto.

Durante la planificación del diseño y desarrollo, la organización debe determinar:

- *Las etapas del diseño y desarrollo.*
- *La revisión, verificación y validación apropiadas para cada etapa del diseño y desarrollo.*
- *Las responsabilidades y autoridades para el diseño y desarrollo.*

La organización debe gestionar las interfaces entre los diferentes grupos involucrados en el diseño y desarrollo para asegurarse de una comunicación eficaz y una clara asignación de responsabilidades.

El diseño y desarrollo debería llevarse a cabo de una manera disciplinada para prevenir o minimizar la ocurrencia de problemas. Este enfoque reduce la dependencia sobre la verificación y validación como únicos métodos para identificar problemas.

Por consiguiente, la organización se debe asegurar de que los productos software se desarrollan conforme a los requisitos especificados y de acuerdo a la planificación de diseño y desarrollo y/o la planificación de calidad.

La planificación de diseño y desarrollo debería tratar lo siguiente, en caso de ser apropiado:

a) las actividades de análisis de requisitos, diseño y desarrollo, codificación, integración, pruebas, instalación y apoyo para la aceptación del producto software.

b) la planificación para el control del producto y provisión del servicio.

c) la organización de los recursos del proyecto.

d) interfaces organizativas y técnicas entre los diferentes individuos o grupos.

e) análisis de posibles riesgos, hipótesis, dependencias y problemas asociados con el diseño y desarrollo.

f) el calendario.

g) la identificación de diferentes normas, métodos, procedimientos y controles.

La planificación debería revisarse periódicamente y modificarse en caso de ser apropiado.

Elementos de entrada para el diseño y desarrollo

Deben determinarse los elementos de entrada relacionados con los requisitos del producto y mantenerse registros. Estos elementos de entrada deben incluir:

- *Los requisitos funcionales y de desempeño.*
- *Los requisitos legales y reglamentarios aplicables.*
- *La información proveniente de diseños previos similares, cuando sea aplicable.*
- *Cualquier otro requisito esencial para el diseño y desarrollo.*

Estos elementos deben revisarse para verificar su adecuación. Los requisitos deben estar completos, sin ambigüedades y no deben ser contradictorios.

Los elementos de entrada para el análisis de los requisitos de software son los requisitos del sistema asignados al software y las especificaciones de los interfaces entre los componentes del sistema.

Los elementos de entrada se pueden determinar de tres formas diferentes, a partir de los requisitos funcionales, de desempeño, de calidad, de seguridad de acceso; y las limitaciones de diseño del sistema, o derivadas de técnicas tales como el prototipado, o a partir de peticiones de cambio de diseño originadas de fases previas en el modelo de desarrollo iterativo, problemas a solucionar, o requisitos surgidos de los criterios de aceptación. Por último, también pueden proceder de las actividades de revisión del contrato.

Resultados del diseño y desarrollo

Los resultados del diseño y desarrollo deben proporcionarse de tal manera que permitan la verificación respecto a los elementos de entrada para el diseño y desarrollo y deben aprobarse antes de su liberación.

Los resultados de diseño y desarrollo deben:

- *Cumplir los requisitos de los elementos de entrada para el diseño y desarrollo.*
- *Proporcionar información apropiada para la compra, la producción y la prestación del servicio.*
- *Contener o hacer referencia a los criterios de aceptación del producto.*
- *Especificar las características del producto que son esenciales para el uso seguro y correcto.*

El resultado del proceso de diseño y desarrollo debería documentarse y definirse de acuerdo con el método elegido o prescrito. Debería ser completo, preciso y coherente con los requisitos. Puede expresarse de diferentes formas, pero es aconsejable utilizar diagramas o una notación de modelos simbólicos que ayude a la interpretación de los resultados.

Los criterios de aceptación para los resultados de diseño y desarrollo deberían definirse con el fin de demostrar que los elementos de entrada a cada fase de diseño y desarrollo están correctamente reflejados en los resultados.

Revisión del diseño y desarrollo

En las etapas adecuadas, deben realizarse revisiones sistemáticas del diseño y desarrollo de acuerdo con lo planificado:

- *Evaluar la capacidad de los resultados de diseño y desarrollo para cumplir los requisitos.*

- *Identificar cualquier problema y proponer las acciones necesarias.*

Los participantes de dichas revisiones deben incluir representantes de las funciones relacionadas con la(s) etapa(s) de diseño y desarrollo que está(n) revisando. Deben mantenerse registros de los resultados de las revisiones y de cualquier acción necesaria.

La organización debería establecer procedimientos para tratar las deficiencias o no conformidades durante las actividades asociadas con el proceso de revisión. Es aconsejable que esos procedimientos se encuentren posteriormente documentados.

Durante las revisiones de diseño y desarrollo, los criterios como viabilidad, seguridad de acceso, seguridad, reglas de programación y ensayabilidad deberían tomarse en cuenta. Además deben llevarse a cabo de acuerdo con las disposiciones planificadas.

Los elementos de la revisión a considerar son los siguientes:

a) qué se va a revisar, cuando y el tipo de revisión.

b) qué grupos funcionales podrían estar afectados en cada tipo de revisión y si hay que celebrar alguna reunión de revisión, como se organizará.

c) qué registros se han de producir.

d) los métodos para el seguimiento de la aplicación de las reglas, prácticas y convenios, para asegurarse de que los requisitos se cumplen.

e) lo que ha de realizarse antes de llevarse a cabo una revisión.

f) lo que ha de hacerse durante la revisión.

g) criterios de éxito de la revisión.

h) las actividades de seguimiento que se han de utilizar para asegurarse de que se han resuelto los aspectos identificados en la revisión.

Verificación del diseño y desarrollo

Se debe realizar la verificación de acuerdo con lo planificado para asegurarse de que los resultados del diseño y desarrollo cumplen los requisitos de los elementos de entrada del diseño y desarrollo. Deben mantenerse registros de los resultados de la verificación y de cualquier acción que sea necesaria.

La verificación debe realizarse de manera adecuada durante el diseño y el desarrollo, con el objetivo de asegurar que la salida de una actividad de diseño y desarrollo es conforme a los requisitos de entrada.

La verificación puede comprender varias revisiones, el resultado del diseño y desarrollo, análisis, demostraciones, simulaciones o pruebas.

Hay en organizaciones que se realizan dos niveles de verificación de una actividad de diseño y desarrollo, una primera revisión del subprograma realizado entre el programador y un superior, como puede ser un analista. Y posteriormente, realizar la verificación entre la organización y la empresa que ha encargado el producto software. Esto suele llevarse a cabo cuando la empresa o cliente no ha sabido transmitir de forma satisfactoria los requisitos de los elementos de entrada del diseño y desarrollo que se tienen que llevar a cabo.

Validación del diseño y desarrollo

Se debe realizar la validación del diseño y desarrollo de acuerdo con lo planificado para asegurarse de que el producto resultante es capaz de satisfacer los requisitos para su aplicación especificada o uso previsto, cuando sea conocido. Siempre que sea factible, la validación debe completarse antes de la entrega o implementación del producto. Deben mantenerse registros de los resultados de la validación y de cualquier acción que sea necesaria.

La validación tiene el objetivo de proporcionar una confianza razonable de que el software cumplirá sus requisitos de operación.

Antes de ofrecer un producto para la aceptación del cliente, conviene que la organización valide la operación del producto de acuerdo con su uso específico, bajo condiciones similares al entorno de aplicación, tal como se especifica en el contrato. Durante la validación cuando sea necesario se pueden realizar auditorías de configuración o evaluaciones, antes de la liberación de una línea de referencia de la configuración. Estas auditorías confirman que el producto software cumple con sus requisitos contractuales o especificados.

Es muy importante registrar los resultados de la validación así como cualquier acción adicional requerida para cumplir los requisitos especificados o contractuales.

En ocasiones no es posible validar totalmente el producto software por mediciones y seguimiento. Por ejemplo, las trazas de seguridad que hay que incorporar a los diferentes productos software, puede ser que solo funciona en el lugar en el que se va a utilizar la aplicación, por lo tanto no podrán validarse hasta que se este probando el producto en el centro del cliente.

Un factor muy importante para desarrollar este proceso, es que la validación a menudo se realiza mediante *pruebas*. Pueden requerirse pruebas a diferentes niveles, desde el elemento de software individual al producto software completo. Existen diferentes enfoques de pruebas, de la extensión de las pruebas y del grado de los controles sobre el entorno de prueba, de las entradas a las pruebas y de las salidas de las pruebas puede variar con el enfoque, la complejidad del producto y el riesgo asociado con la uso del producto.

Las pruebas específicas del software incluyen el establecimiento, la documentación, la revisión e implementación de los planes para lo siguiente:

a) pruebas unitarias, por ejemplo pruebas únicas para componentes software.

b) pruebas de integración y del sistema, por ejemplo pruebas de componentes software agregados.

c) pruebas de calificación, por ejemplo pruebas del producto software completo antes de la entrega para confirmar que el software cumple los requisitos definidos.

d) pruebas de aceptación, por ejemplo pruebas del producto software completo para confirmar que el software cumple los criterios de aceptación.

Control de los cambios del diseño y desarrollo

Los cambios del diseño y desarrollo deben identificarse y deben mantener los requisitos. Los cambios deben revisarse, verificarse y validarse, según sea apropiado, y aprobarse antes de su implementación. La revisión de los cambios del diseño y desarrollo debe incluir la evaluación del efecto de los cambios en las partes constitutivas y en el producto ya entregado.

Deben mantenerse registros de los resultados de la revisión de los cambios y de cualquier acción que sea necesaria.

En el entorno de desarrollo software, el control de los cambios al diseño y desarrollo se trata habitualmente como parte de la gestión de la configuración.

Conviene que los cambios a una especificación software o componente mantengan la adecuada consistencia entre los requisitos, diseño, código, especificaciones de pruebas, manuales de usuario…

Producción y prestación del servicio

Control de la producción y de la prestación del servicio

La organización debe planificar y llevar a cabo la producción y la prestación del servicio bajo condiciones controladas. Las condiciones controladas deben incluir, cuando sea aplicable:

- *La disponibilidad de información que describa las características del producto.*
- *La disponibilidad de instrucciones de trabajo, cuando sea necesario.*
- *El uso del equipo apropiado.*
- *La disponibilidad y uso de dispositivos de seguimiento y medición.*
- *La implementación del seguimiento y de la medición.*
- *La implementación de actividades de liberación, entrega y posteriores a la entrega.*

- Producción y prestación del servicio software

Un proyecto software se debería organizar de acuerdo a un conjunto de procesos, que transforme los requisitos en un producto software. Los requisitos de *"control de producción y prestación del servicio"* especificados en la norma anterior, equivalen para productos software para:

a) actividades de versión, por ejemplo construcción, versión y replicación.

b) actividades de entrega, por ejemplo entrega e instalación.

c) actividades de después de la entrega, por ejemplo mantenimiento del producto.

- Construcción y versión

Es aconsejable activar procesos para construir, versionar y replicar los elementos software.

Las siguientes provisiones son adecuadas para la construcción y versión:

a) identificación de los elementos software que componen cada versión.

b) identificación de los tipos de versión, dependiendo de la frecuencia o impacto sobre las operaciones del cliente y habilidad para implementar los cambios.

c) criterios de decisión y guía para determinar donde las correcciones temporales localizadas pueden incorporarse.

- Replicación

Cuando es necesario, conviene que la organización realice una replicación, teniendo en cuenta:

a) identificación del archivo maestro y las copias.

b) el tipo de medios para cada elemento software.

c) estipulación de la documentación requerida.

d) control del entorno bajo el que se realiza la replicación para asegurar la repetibilidad.

e) provisión para garantizar la corrección.

- Entrega

La entrega se puede realizar mediante el movimiento físico de los medios o por la transmisión electrónica.

- Instalación

En ocasiones, el cliente lleva a cabo la instalación. En dicho caso el papel de la organización es descubrir los pasos que el cliente necesita tomar para llevar a cabo la instalación. En otras ocasiones es la propia organización la que realiza la instalación.

- Operaciones

La organización que desarrolla el software debería organizar y controlar las operaciones, teniendo en cuenta la necesidad de establecer un apoyo al usuario y disposiciones para garantizar la continuidad del apoyo.

- Mantenimiento

El contrato debería contemplar el mantenimiento del producto software que se pida por el cliente para elementos específicos, y un periodo específico de tiempo, después de la entrega inicial y la instalación. Un proceso común que suelen desempeñar las organizaciones para llevar a cabo el mantenimiento y su posterior verificación, es llevar a cabo el mantenimiento del producto software a través de la recepción de incidencias, en la que se incluirá toda la documentación necesaria.

Validación de los procesos de la producción y de la prestación del servicio

La organización debe validar aquellos procesos de producción y de prestación del servicio donde los productos resultantes no puedan verificarse mediante actividades de seguimiento o medición posteriores. Esto incluye a cualquier proceso en el que las deficiencias se hagan aparentes únicamente después de que el producto esté siendo utilizado o se haya prestado el servicio.

La validación debe demostrar la capacidad de estos procesos para alcanzar los resultados planificados.

La organización debe establecer las disposiciones para estos procesos, incluyendo, cuando sea aplicable:

- *Los criterios definidos para la revisión y aprobación de los procesos.*
- *La aprobación de equipos y calificación del personal.*
- *El uso de métodos y procedimientos específicos.*
- *Los requisitos de los registros y la revalidación.*

La organización debería considerar que procesos pueden utilizarse para compensar la falta de habilidad para validar totalmente el producto. Ejemplos:

- una revisión del diseño y del desarrollo puede considerar como el diseño y el desarrollo puede fallar en adición a la comprobación más sencilla de que el diseño y el desarrollo funcionarán correctamente.

- un programa en modo de fallo y análisis de efecto que describe el historial de fallos de diseño y desarrollo, y como se pueden evitar.

Identificación y trazabilidad

Cuando sea apropiado, la organización debe identificar el producto por medios adecuados, a través de toda la realización del producto.

La organización debe identificar el estado del producto con respecto a los requisitos de seguimiento y medición.

Cuando la trazabilidad sea un requisito, la organización debe controlar y registrar la identificación única del producto.

La identificación y la trazabilidad se implantan comúnmente a través de la gestión de la configuración.

La gestión de la configuración es una disciplina de gestión que aplica la dirección técnica y administrativa al diseño, desarrollo y apoyo a los elementos de configuración, incluyendo elementos software. Su objetivo principal es la de proporcionar una total visibilidad de la configuración y el estado presente del producto.

- Trazabilidad

A lo largo del ciclo de vida del software debería haber un proceso para trazar los componentes de un elemento o un producto software. Su ámbito puede variar de acuerdo con los requisitos del contrato o el mercado.

Propiedad del cliente

La organización debe cuidar los bienes que son propiedad del cliente mientras estén bajo el control de la organización o estén siendo utilizados por la misma. La organización debe identificar, verificar, proteger y salvaguardar los bienes que son propiedad del cliente suministrados para su utilización o incorporación dentro del producto. Cualquier bien que sea propiedad del cliente que se pierda, deteriore o que de algún otro se considere inadecuado para su uso debe ser registrado y comunicado al cliente.

Debería definirse que actualizaciones de elementos suministrados por el cliente se deberían aceptar e integrar. La organización puede aplicar los mismos tipos de actividades de verificación al producto suministrado por el cliente que al producto adquirido.

Anexo

Además del estudio de los puntos que se han expuesto en la norma ISO-9001:2000, se ha considerado necesario incluir el siguiente punto que no se había tratado anteriormente, ya que la medición, el análisis y mejora es un factor importante con el cual se puede demostrar que los productos de una organización poseen la calidad acordada con el cliente.

Medición, análisis y mejora

La organización debe planificar e implementar los procesos de seguimiento, medición, análisis y mejora necesarios para:

- *demostrar la conformidad del producto.*
- *asegurarse de la conformidad del sistema de gestión de la calidad.*
- *mejorar continuamente la eficacia del sistema de gestión de la calidad.*

Esto debe comprender la determinación de los métodos aplicables, incluyendo técnicas estadísticas, y el alcance de su utilización.

El propósito de la medición es la recogida, el análisis y la información de datos relativos a los productos desarrollados y a los procesos implementados en la unidad organizativa, para apoyar la gestión efectiva de los procesos y demostrar de forma objetiva la calidad de los productos.

Satisfacción del cliente

Como una de las medidas del desempeño del sistema de gestión de la calidad, la organización debe realizar el seguimiento de la información relativa a la percepción del cliente con respecto al cumplimiento de sus requisitos por parte de la organización. Deben determinarse los métodos para obtener y utilizar dicha información.

El proceso de la organización para solicitar, medir y hacer seguimiento de la respuesta de satisfacción del cliente debería proveer información sobre una base continua o periódica, según sea apropiado.

Por ejemplo, para el desarrollo de software se puede considerar:

- métricas de calidad en uso derivadas de la respuesta directa e indirecta del cliente.

- número de versiones de software necesarias para solucionar los problemas, después de la entrega inicial.

Auditoría interna

La organización debe llevar a cabo a intervalos planificados auditorías internas para determinar si el sistema de gestión de la calidad:

- *es conforme con las disposiciones planificadas con los requisitos de esta norma internacional y con los requisitos del sistema de gestión de la calidad establecidos por la organización.*

- *se ha implementado y se mantiene de forma eficaz.*

Se debe planificar un programa de auditorías tomando en consideración el estado y la importancia de los procesos y las áreas a auditar, así como los resultados de auditorías previas. Se deben definir los criterios de auditoría, el alcance de la misma, su frecuencia y metodología. La selección de los auditores y la realización de las auditorías deben asegurar la objetividad e imparcialidad del proceso de auditoría. Los auditores no deben auditar su propio trabajo.

Deben definirse, en un procedimiento documentado, las responsabilidades y requisitos para la planificación y la realización de auditorías, para informar de los resultados y para mantener los registros.

La dirección responsable del área que esté siendo auditada debe asegurarse de que se toman acciones sin demora injustificada para eliminar las no conformidades detectadas y sus causas. Las actividades de seguimiento deben incluir la verificación de las acciones tomadas y el informe de los resultados de la verificación.

Cuando las organizaciones de software separan su trabajo por proyectos, la planificación de la auditoría debería definir una selección de proyectos y evaluar tanto el cumplimiento de su planificación de calidad de proyecto respecto al sistema de gestión de calidad de la organización, como el cumplimiento del proyecto respecto a la planificación de calidad del proyecto.

Esto puede requerir que la auditoría de varios proyectos en estados distintos de su ciclo de vida de desarrollo del producto, o la auditoría de un único proyecto según avanza por varios estados. En dónde el proyecto objetivo cambia su programación temporal, puede revisarse la programación de la auditoría interna, bien para cambiar el calendario de la auditoría o bien para considerar un proyecto distinto.

Capítulo 3: Introducción a Oracle

Concepto de base de datos

Una base de datos es un sistema para almacenar un conjunto de datos pertenecientes a un mismo contexto de manera tal que los datos que la conforman puedan ser utilizados en forma fragmentada cuando sea necesario. En este sentido, una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos indexados para su posterior consulta. Hoy día, debido al desarrollo tecnológico de campos como la informática y la electrónica, la mayoría de las bases de datos están en formato digital.

Tradicionalmente, se suelen organizar según campos, registros o archivos. El campo es una fracción única de información; el registro, un sistema completo de campos y el archivo, un conjunto o colección de registros.

CD_PERSONA	DS_NOMBRE	DS_APELLIDO1	DS_APELLIDO2
01	Dan	Brown	
02	Ken	Follet	
03	Carlos	Ruiz	Zafón

Campo

Registro

Archivo

Tabla 3. Representación de una tabla de base de datos

Existen tanto bases estáticas como dinámicas. Las primeras son sólo de lectura y que generalmente se utilizan para almacenar datos históricos que podrán ser utilizados a lo largo del tiempo para, por ejemplo, realizar proyecciones. Las dinámicas son las que contienen información que puede ser modificada tanto para actualizar los datos que la integran como para agregar nuevos.

El modelo de base de datos actual más utilizado es la *base de datos relacional*, esta permite modelar problemas reales y administrar datos de forma dinámica. Sus fundamentos fueron postulados por Edgar Frank Codd en 1970, en poco tiempo se consolidó como un nuevo paradigma en los modelos de las bases de datos. Su idea fundamental es el uso de relaciones (un conjunto de tuplas donde todas tienen los mismos atributos).

En este modelo, el lugar y la forma en que se almacenan los datos no tiene relevancia, por lo que es más fácil de entender y de utilizar para un usuario esporádico de base de datos. La información puede ser recuperada o almacenada mediante consultas que ofrecen una amplia flexibilidad y poder para administrar la información.

El lenguaje más habitual para construir las consultas a bases de datos relacionales es el *SQL* (Structured Query Language o Lenguaje Estructurado de Consultas), un estándar implementado por los principales motores o sistemas de gestión de bases de datos relacionales.

Personas que interactúan con una base de datos

A la hora de definir las personas que interactúan con una base de datos, se debe diferenciar entre aquellas personas que van a emplear la base de datos y las que están relacionadas con el diseño, elaboración y funcionamiento del software.

En el primer grupo se debe destacar a los administradores de bases de datos, diseñadores de bases de datos y usuarios finales.

Administrador de base de datos: Su función es la vigilancia y gestión de los datos. Debe asegurarse que los datos no se destruyan ni se corrompan, permitiendo su confidencialidad, disponibilidad e integridad. También será el responsable de establecer el sistema de autorizaciones de acceso y deberá coordinar y controlar su uso.

Diseñador de base de datos: Identificará que datos se almacenarán en la base de datos y que estructura será la adecuada para presentar y almacenar dichos datos. Estas tareas suelen realizarse antes de la implementación de la base de datos.

Debe comunicarse con los futuros usuarios de la base de datos para conocer sus necesidades y requisitos, de manera que pueda desarrollar una base de datos que satisfaga dichas necesidades y requisitos.

Usuario final: El usuario final es aquel que tiene que acceder a los datos porque los necesita para llevar a cabo su actividad (consulta, actualización de datos...). Su interés suele estar únicamente centrado en los datos.

En el segundo grupo debe destacarse a los desarrolladores y personal de mantenimiento.

Desarrolladores: Personas que diseñan e implementan los *paquetes software* que facilitan el diseño y utilización del sistema. (Paquetes para el diseño y elaboración de bases de datos, prototipos, simuladores…)

Personal de mantenimiento: Su labor es el seguimiento del funcionamiento y mantenimiento real del entorno hardware y software de la base de datos.

Sistema gestor de base de datos

Los *sistemas de gestión de base de datos* (SGBD) son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Este software debe permitir crear y gestionar una base de datos asegurando su integridad, confidencialidad y seguridad, con el objetivo de proporcionar un entorno que sea eficiente a la vez que conveniente para ser utilizado al extraer y almacenar información de la base de datos. Todo SGBD debe permitir crear, manipular y construir una base de datos.

El propósito general de un SGBD es el de manejar de manera clara, sencilla y ordenada un conjunto de datos. Para llevar a cabo este propósito se deben cumplir una serie de objetivos:

- ***Abstracción de la información:*** Los SGBD ahorran a los usuarios detalles acerca del almacenamiento físico de los datos. Da lo mismo si una base de datos ocupa uno o cientos de archivos, este hecho se hace transparente al usuario.

- ***Independencia:*** La independencia de los datos consiste en la capacidad de modificar el esquema (físico o lógico) de una base de datos sin tener que realizar cambios en las aplicaciones que se sirven de ella.

- ***Redundancia mínima:*** Un buen diseño de una base de datos logrará evitar la aparición de información repetida o redundante.

- ***Consistencia:*** En aquellos casos en los que no se ha logrado esta redundancia nula, será necesario vigilar que aquella información que aparece repetida se actualice de forma coherente, es decir, que todos los datos repetidos se actualicen de forma simultánea.

- ***Seguridad:*** La información almacenada en una base de datos puede llegar a tener un gran valor. Los SGBD deben garantizar que esta información se encuentra segura frente a usuarios malintencionados, que intenten leer información privilegiada; frente a ataques que deseen manipular o destruir la información. Normalmente, los SGBD disponen de un complejo sistema de permisos a usuarios y grupos de usuarios, que permiten otorgar diversas categorías de permisos.

- ***Integridad:*** Se trata de adoptar las medidas necesarias para garantizar la validez de los datos almacenados. Es decir, se trata de proteger los datos ante fallos de hardware, datos introducidos por usuarios descuidados, o cualquier otra circunstancia capaz de corromper la información almacenada.

- ***Respaldo y recuperación:*** Los SGBD deben proporcionar una forma eficiente de realizar copias de respaldo de la información almacenada en ellos, y de restaurar a partir de estas copias los datos que se hayan podido perder.

- ***Control de la concurrencia:*** En la mayoría de entornos lo más habitual es que sean muchas las personas que acceden a una base de datos, bien para recuperar información, bien para almacenarla. Y es también frecuente que dichos accesos se realicen de forma simultánea. Así pues, un SGBD debe controlar este acceso concurrente a la información, que podría derivar en inconsistencias.

- ***Tiempo de respuesta:*** Lógicamente, es deseable minimizar el tiempo que el SGBD tarda en darnos la información solicitada y en almacenar los cambios realizados.

Ventajas e inconvenientes de un SGBD

Ventajas

- ***Control de redundancia de datos.*** En los sistemas de archivos tradicionales se almacenaban los datos en más de un archivo por lo que se desperdiciaba espacio, en los SGBD se controla dicha redundancia siendo solamente necesaria la duplicidad de datos en algunos casos.

- ***Coherencia de los datos.*** Al controlar la redundancia de datos, se controla también la coherencia de los mismos ya que las actualizaciones que se realicen sobre ellos se realizarán una sola vez.

- ***Compartición de datos.*** Las bases de datos pertenecen a una organización y debe ser utilizada y compartida por todos sus miembros siempre y cuando tengan los permisos necesarios para hacerlo, en los sistemas de archivos estos pertenecían a la persona que los utilizaba.

- ***Mayor integridad de los datos.*** Una base de datos debe tener restricciones, es decir, reglas que no se puedan violar y que garantizan la coherencia y validez de los datos.

- ***Mayor seguridad.*** Todos los datos de la base de datos deben estar protegidos ante el uso por parte de personal no autorizado, esta seguridad es impuesta por SGBD en sí, así como por el administrador que es el encargado de asignar a los usuarios los diferentes permisos sobre la base de datos.

- ***Imposición de estándares.*** Los SGBD permiten al administrador imponer estándares.

- ***Economía de escala.*** A diferencia de los sistemas de archivos la centralización de la base de datos disminuye enormemente los costes.

- ***Mejor accesibilidad a los datos y mayor capacidad de respuesta.*** Los datos son accesibles directamente a los usuarios finales.

- ***Productividad mejorada.*** Reducción del tiempo del tiempo de desarrollo al proporcionar el SGBD todas las rutinas de bajo nivel.

- ***Mantenimiento más sencillo gracias a la independencia de los datos.*** El SGBD separa los datos de las aplicaciones por lo que gracias a esta independencia se simplifica el mantenimiento.

- ***Mayor nivel de concurrencia.*** Los SGBD gestionan el nivel de concurrencia de la base de datos aportando mecanismos suficientes para su control.

- ***Servicios mejorados de copia de seguridad y recuperación.*** Ante la posibilidad de posibles fallos y por consiguiente la pérdida de información los SGBD aportan facilidades para disminuir la cantidad de procesamiento que se pierde por cada fallo.

Inconvenientes

- ***Complejidad.*** Debido a las grandes ventajas enumeradas, además de otras funcionalidades que debe aportar, es inevitable que el sistema sea de gran complejidad.

- ***Tamaño.*** Gran ocupación de tamaño en disco duro así como requerimiento de gran cantidad de memoria para su eficiencia.

- ***Costes del SGBD.*** En los sistemas gestores de base de datos multiusuario el precio se eleva enormemente pudiendo variar entre 100.000 y 1.000.000€.

- ***Costes del hardware adicional.*** Debido al tamaño y complejidad es necesaria la utilización de hardware capaz de soportarlo de manera eficiente.

- ***Prestaciones.*** Habitualmente los SGBD son escritos para su utilización genérica para varias aplicaciones, por lo que en los sistemas de archivos al construirse para una sola aplicación las prestaciones suelen ser mucho mejores.

- ***Mayor impacto de los fallos.*** Debido a la gran utilización de una base de datos tanto de las aplicaciones como de los usuarios, un posible fallo puede que detenga todas las demás operaciones.

Estructura de un SGBD

En la siguiente imagen se muestra la estructura que tiene un sistema gestor de bases de datos.

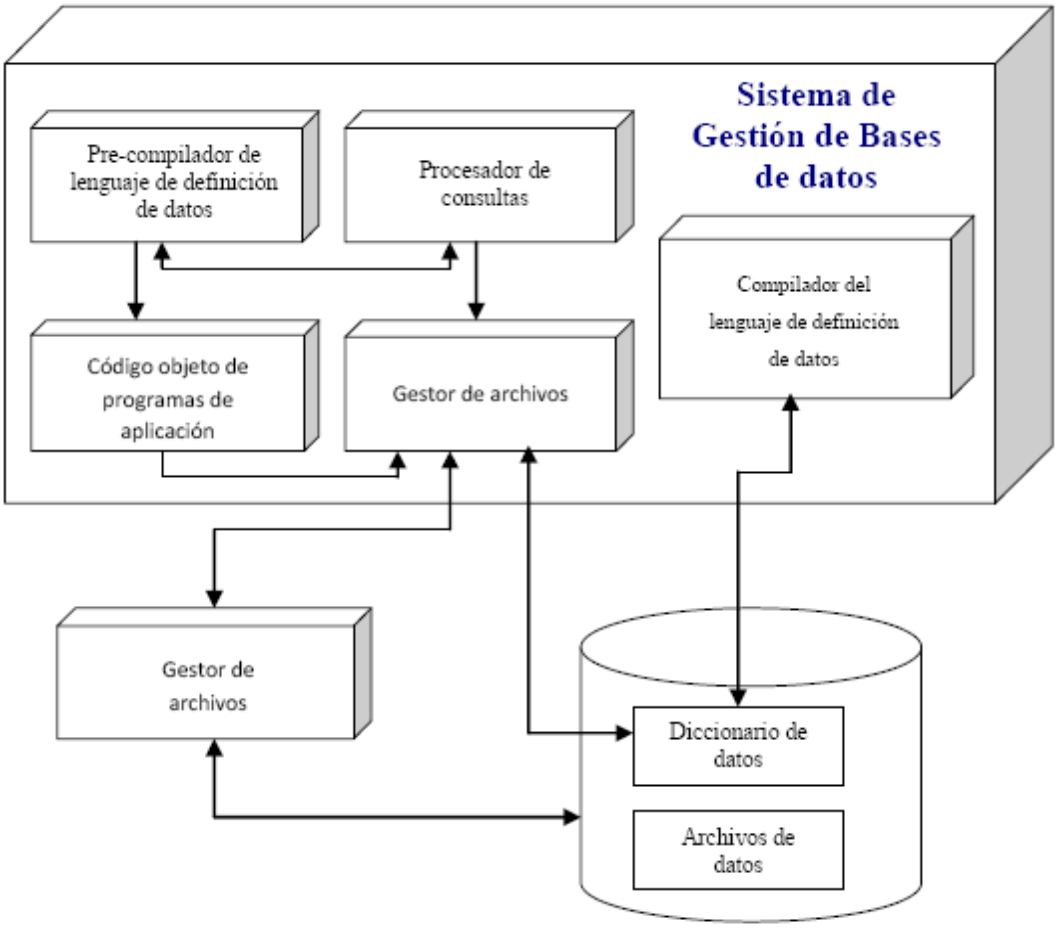

Imagen 3. Componentes del Sistema Gestor de Base de Datos

- ***Gestor de archivos:*** Gestiona la asignación de espacio en disco y de las estructuras de datos utilizadas para la representación de la información almacenada en él.

- ***Gestor de la base de datos:*** Proporciona la interfaz entre los datos de bajo nivel almacenados en la base de datos y los programas de la aplicación y las consultas que se hacen en el sistema. Determina que registros son necesarios para satisfacer la petición y realiza una llamada al gestor de archivos para ejecutarla.

- ***Gestor de diccionario de datos:*** Controla los accesos al diccionario de datos y se encarga de mantenerlo. Almacena *metadatos* sobre la estructura de la base de datos.

- ***Procesador de consultas:*** Transforma las consultas en un conjunto de instrucciones de bajo nivel que se dirigen al gestor de la base de datos.

- ***Precompilador del LMD (Lenguaje de Manipulación de Datos):*** Convierte las sentencias LMD en llamadas a funciones estándar escritas en el lenguaje principal. Trabaja con el procesador de consultas para generar el código apropiado.

- ***Compilador del LDD (Lenguaje de Definición de Datos):*** Convierte las sentencias del LDD en un conjunto de tablas de metadatos que son almacenadas en el diccionario de datos.

Historia de Oracle

Como se expuso en el capítulo de introducción del proyecto, Oracle surge a finales de los años setenta bajo el nombre de Relational Software a partir de un estudio sobre SGBD de George Koch. Computer World definió este estudio como uno de los más completos jamás escritos sobre bases de datos. Este artículo incluía una comparativa de productos que erigía a Relational Software como el más completo desde el punto de vista técnico. Esto se debía a que usaba la filosofía de las bases de datos relacionales, algo que por aquella época era todavía desconocido.

Larry Ellison vio una oportunidad que otras empresas no supieron apreciar al descubrir la descripción de un prototipo de trabajo para una base de datos relacional y enterarse de que ninguna se había comprometido a comercializar la tecnología. Ellison y sus cofundadores, Bob Miner y Ed Oates, se dieron cuenta del gran potencial económico que ofrecía el modelo de base de datos relacional, aunque nunca supusieron que cambiarían la informática empresarial para siempre.

Para ver cómo ha ido evolucionando Oracle a lo largo del tiempo se expondrá el siguiente cronograma, en el que se resaltan los hitos más importantes en la evolución de Oracle.

1970 – El Dr. Edgar Frank Codd expuso su teoría de modelo de datos relacionales.

1977 – Larry Ellison, Bob Miner y Ed Oates constituyen *"Software Development Laboratories"* (SDL) con $2,000. Firman un contrato con la CIA para desarrollar un programa de base de datos cuyo nombre es Oracle. SDL finaliza el proyecto para la CIA un año antes de lo previsto y utiliza ese año para desarrollar el primer sistema de gestión de base de datos relacional del mercado.

1978 – SDL cambia su nombre a Relational Software Inc. (RSI). Los tres fundadores comparten una visión de software portátil que sea compatible con el lenguaje estructurado de consulta (SQL) de IBM y los mini computadores del mercado.

1979 – RSI saca la primera versión, versión 2 (no hay una versión 1 creyendo que la gente no compraría la primera versión de un producto) de la base de datos escrita en lenguaje ensamblador. La primera versión comercial del software es vendida a la base de la fuerza aérea Wright-Patterson. Es la primera base de datos comercial, RDBMS (Relational Data Base Management System o Sistema de gestión de base de datos relacional), en el mercado.

1981 – La primera herramienta, Interactive Application Facility (IAF), que es un predecesor del futuro SQL*Forms, es creada. Esta herramienta permite al usuario desarrollar aplicaciones para la recuperación de datos en informes más fáciles de usar.

1982 – RSI cambia su nombre a Oracle Systems Corporation (OSC), posteriormente se simplifica el nombre a Oracle Corporation.

1983 – La versión 3, desarrollada en el lenguaje de programación *C* es vendida. Bob Miner se encarga de desarrollar la mitad, mientras también mantiene la versión 2 basada en ensamblador, y Bruce Scott desarrolla la otra mitad. Es el primer RDBMS a 32-bits.

1984 – Se lanza la versión 4. Las primeras herramientas son lanzadas (IAG-genform, IAG-runform, RPT). La primera base de datos con consistencia en la lectura. Oracle es migrado al PC.

1985 – La versión 5 y 5.1 son lanzadas. Se crea el primer Parallel Server sobre VMS/VAX.

1986 – Oracle se abre al público el 12 de Marzo. La acción se abre a $15 y se cierra a $20.75. El servicio cliente / servidor de Oracle es introducido; es la primera base de datos cliente servidor.

1987 – Oracle es la compañía más grande de DBMS. El grupo de aplicaciones Oracle inicia su trabajo. Oracle se convierte en la primera base de datos SMP (symmetrical multi-processing).

1988 – Se lanza la versión 6 de Oracle. Se ofrece el primer bloqueo a nivel de registros, primer backup en caliente, además de introducirse PL/SQL.

1989 – Oracle introduce el soporte OLTP (Online Transaction Processing o Procesamiento de Movimientos en línea), es una clase de programa que facilita los programas orientados a movimientos. OLTP es una de las piezas angulares para el desarrollo de negocios online, como puede ser la banca o el comercio electrónico.

1992 – Oracle lanza la versión 7.

1993 – Las herramientas gráficas, cliente servidor son introducidas. Las aplicaciones de Oracle se mueven de ambiente carácter a modo cliente servidor.

1994 – Bob Miner, el genio detrás de Oracle muere de cáncer.

1995 – Oracle desarrolla la primera base de datos de 64-bits.

1996 – Oracle lanza la versión 7.3.

1997 – Se introduce la versión 8. Aparece Oracle Application Server. Se introducen los aplicativos Web. Oracle es la primera base de datos Web. Las herramientas de BI como Discoverer son introducidas para el almacenamiento de datos. Las herramientas poseen soporte nativo para Java.

1998 – Oracle es portado a Linux. Se libera Applications 11. Oracle es la primera base de datos con soporte para XML.

1999 – Aparece Oracle 8i. Integra Java/XML en las herramientas de desarrollo.

2001 – Larry Ellison anuncia el lanzamiento de la base de datos Oracle 9i, una solución de gestión de datos, según Larry Ellison, que revolucionaría la economía de la computación. Ésta entre otras características podrá ejecutarse en paquetes de aplicaciones que usen tanto la base de datos en Unix como en Windows.

2002 – Este año incluye el lanzamiento de nuevas versiones Oracle en las tres principales líneas de productos, Oracle Database, Oracle Application Server y Oracle E-Business Suite. Con el nuevo lanzamiento de Oracle 9i, Oracle introduce el primer soporte nativo para base de datos relacionales en XML.

2003 – Oracle 10g es lanzada al mercado, la primera base de datos diseñada para ejecutarse en una red de ordenadores de bajo costo. Además Oracle lanza Oracle Application Server 10g, el primer software que simplifica la gestión de aplicaciones que se ejecutan en un entorno de computación distribuida.

2005 – Oracle presenta su base de datos gratuita para desarrolladores, estudiantes..., Oracle Database 10g Express Edition. Esta edición permite a las pequeñas empresas beneficiarse de este software de forma gratuita, y así obtener ventaja sobre sus competidores al mismo tiempo que reducen costos.

2006 – Oracle es la primera en apoyo, administración de seguridad, copias de seguridad, gestión de contenidos, y Linux.

2007 – Oracle 11g es anunciada, introduce Oracle Flashback Data Archive. Esta nueva función transparente sigue la pista a los cambios de todos los datos de Oracle en un medio altamente seguro y eficiente. Es la primera base de datos en incluir captura y reproducción de datos para Oracle Real Application Testing.

Estructura de la base de datos Oracle

La base de datos Oracle tiene dos estructuras primarias: una estructura física, que hace referencia a los datos realmente almacenados en cintas magnéticas, discos y otros; y una estructura lógica, correspondiente a una representación abstracta de los datos almacenados. La base de datos contiene los siguientes tipos de ficheros de datos:

- ***Ficheros de datos (data files):*** contienen el diccionario de datos, objetos de usuario e imágenes anteriores de datos que son modificadas por transacciones actuales.

- ***Ficheros de diario:*** Denominados ficheros de diario para rehacer (redo log). Estos ficheros registran todos los cambios realizados sobre los datos y se utilizan en el proceso de recuperación, si los cambios realizados no se almacenan en memoria permanente.

- ***Ficheros de control:*** Contienen la información de control: nombre de base de datos, nombre de ficheros, ubicación de ficheros,... Este fichero también se necesitará en el caso de tener operaciones de recuperación.

- ***Ficheros de traza y diario de alertas:*** Los procesos background tienen un fichero de traza asociado a ellos y el diario de alertas lleva un seguimiento de los eventos principales de la base de datos.

La instancia Oracle consta de un área Global del Sistema y un conjunto de procesos background y consta de los siguientes componentes:

- ***Área Global del Sistema (SGA):*** Esta área de memoria se usa para almacenar la información de la base de datos (información de control y datos de la instancia) compartida por los usuarios. Oracle asigna una SGA cuando comienza una instancia. El SGA se divide en:

    - **Búfer de la base de datos en caché:** contiene los bloques de datos de la base de datos más reciente accedidos o utilizados por los usuarios.

    - **Búfer del diario:** para rehacer también llamado búfer de redo log, que es una zona donde se almacena de forma secuencial toda la información de los cambios realizados a la base de datos (*insert*, *update*, *delete*, *create*, *alter* o *drop*) y se usa para la recuperación.

    - **Caché de biblioteca:** en esta área se encuentran las sentencias SQL que han sido analizadas.

    - **Caché de diccionario de datos:** se usa para almacenar la información de los datos más recientes utilizada, como definiciones de tablas y columnas, nombres de usuario, claves y privilegios.

    - **Conjunto grande o large pool:** es un área opcional de memoria de la SGA que se utiliza cuando las transacciones interactúan con más de una base de datos.

    - **Conjunto Java o Java pool:** se necesita si se instala y se utiliza Java ya que realiza el análisis de servicios para comandos Java.

- ***Procesos de Usuario:*** Cada proceso del usuario corresponde a la ejecución de alguna aplicación o a alguna otra herramienta.

- ***Área global de Programa (PGA):*** Es un búfer de memoria que contiene datos e información de control para un proceso del servidor. Oracle crea un PGA cuando se comienza un proceso del servidor.

- ***Procesos de Oracle:*** Un proceso es un mecanismo de un sistema operativo que puede ejecutar una serie de pasos. Un proceso tiene su propia área de memoria privada donde se ejecuta.

  Oracle crea sus procesos de servidor para manipular las peticiones de procesos de usuarios conectados.

  Los procesos background se crean para cada instancia de Oracle; realizan la E/S de una manera asíncrona y proporcionan paralelismo para un mejor rendimiento y fiabilidad. Son programas para mantener y mejorar la transferencia entre las estructuras física y la memoria.

Arranque y parada de Oracle

Arranque de la base de datos

La base de datos en Oracle no estará disponible para ser utilizada hasta que ésta no haya sido abierta. Hay diferentes opciones a la hora de abrir la base de datos, en caso de no especificar ninguna opción para llevar a cabo esta operación deberán seguirse los siguientes pasos:

1. ***Iniciar una instancia de la base de datos:*** Se asignará la SGA y se crean los procesos en background. En el fichero de inicialización de parámetros se leerá la información necesaria (nombre de la base de datos, memoria necesaria, nombre de los ficheros de control,...).

2. ***Montar una base de datos:*** En este momento se asociará una instancia a la base de datos, se buscarán los ficheros de control donde se leerá información como: nombre de fichero de datos... Cuando la base de datos solamente se encuentra montada sólo está accesible para el administrador de la misma.

3. ***Apertura de la base de datos:*** Se abren los ficheros necesarios: ficheros de datos... A partir de este momento los usuarios podrán tener acceso a la base de datos.

A continuación se enumerarán las diferentes opciones que se pueden tener en cuenta a la hora de abrir una base de datos.

- ***FORCE:*** Abre la base de datos si está cerrada y si está abierta se cierra y vuelve a abrirla otra vez.

- ***NOMOUNT:*** Con esta opción se inicia la instancia sin montar la base de datos.

- ***MOUNT:*** Con esta opción se inicia la instancia y se monta la base de datos sin abrirla.

- ***DBA:*** Es una opción de acceso restringido, es decir, sólo se permite el acceso al administrador de la base de datos.

- ***PFILE = 'nombre fichero':*** Esta opción permite indicar el nombre del fichero de inicialización de parámetros.

Parada de la base de datos

Para parar la base de datos habrá que realizar las mismas operaciones que para arrancarla pero en sentido inverso.

1. ***Cerrar base de datos.***

2. ***Desmontar la base de datos.***

3. ***Parar la instancia Oracle.***

Al igual que en el proceso de arranque de la base de datos, a continuación se enumerarán las diferentes opciones que pueden tenerse en cuenta a la hora de parar una base de datos.

- ***NORMAL:*** Opción por defecto de Oracle. Cuando el administrador cierra la base de datos mediante esta opción no se permitirán nuevas conexiones pero se esperará hasta que el último usuario se desconecte por lo que no será necesario recuperar datos cuando se vuelva a arrancar la base de datos.

- ***ABORT:*** Con esta opción el administrador fuerza a que todas las sentencias SQL en proceso se terminen inmediatamente y a que se desconecten instantáneamente todos los usuarios. Aquellas transacciones que no han sido realizadas no se recuperan hasta el siguiente arranque.

- ***IMMEDIATE:*** Con esta opción el administrador permite que se termine el proceso de las sentencias SQL, pero desconectará inmediatamente a todos los usuarios y las transacciones que no han sido realizadas no se recuperan hasta el siguiente arranque.

- ***TRANSACTIONAL:*** El administrador permitirá que termine el proceso de las transacciones pero no se permitirá iniciar ninguna nueva. Se desconectarán todos los usuarios después de que todas las transacciones se hayan realizado o se hayan deshecho.

Estructura de una base de datos

Estructura física

La estructura física de una base de datos está compuesta por el bloque de datos y archivos o ficheros.

- ***Bloque de datos:*** el bloque de datos representa la unidad más pequeña de entrada/salida. Su tamaño normalmente sería un múltiplo del tamaño del bloque del sistema operativo.

- ***Archivos o ficheros:*** además de los ficheros que se han comentado con anterioridad, se pueden encontrar otros, estos son:

    - ***Archivo de clave:*** utilizado para autentificar los usuarios privilegiados de la base de datos.

    - ***Archivo de parámetro:*** utilizado para definir las características de una instancia Oracle.

    - ***Archivo redo log archivados:*** fichero que contiene copias fuera de control de los ficheros redo log que pueden ser necesarios para la recuperación al producirse errores.

Estructura lógica

La estructura lógica de una base de datos está compuesta por el tablespace, extensiones y diversos segmentos.

- ***Tablespaces:*** almacenan los datos de la base de datos y están formados por uno o más archivos de datos.

- ***Extensiones:*** están formadas por una serie de bloques contiguos. Cuando se crea un objeto se reserva una extensión y cuando el objeto crezca, al necesitar más espacio, será necesario reservar más extensiones.

- ***Segmentos:*** Un segmento se define como una serie de extensiones, como mínimo contendrá una extensión. Los diferentes tipos de segmentos son:

    - ***Segmentos de datos:*** almacenará todos los datos de una tabla. Estos segmentos se crearán automáticamente al crear una tabla.

    - ***Segmentos de índices:*** almacenará todos los datos de un índice. Estos segmentos se crearán automáticamente al crear el índice.

    - ***Segmentos temporales:*** se utilizan para procesar consultas cuando no hay espacio en la memoria para realizar esta operación.

    - ***Segmentos de Rollback:*** se almacena la diferente información de los datos que han sido cambiados por transacciones.

Diccionario de datos

El diccionario de datos está compuesto por un conjunto de tablas y vistas asociadas donde se almacena toda la información sobre los objetos que componen la base de datos, así como la estructura lógica y física de la misma.

El diccionario de datos incluye dos tipos de objetos: tablas base y vistas.

- Las tablas base se crean automáticamente cuando se crea la base de datos con el comando "*create database*", y son las que realmente contienen la información del diccionario de datos.

- Las vistas se crean al lanzar el script "*catalog.sql*" y permite acceder a la información de las tablas del diccionario de datos.

Éste contiene información sobre la definición de todos los objetos de la base de datos (tablas, vistas, índices, sinónimos, secuencias, procedimientos, funciones, paquetes, triggers, etc), el espacio ocupado por cada objeto, condiciones de integridad, usuarios, privilegios, roles, así como auditorías del sistema.

Capítulo 4: Calidad y seguridad en Oracle

Introducción

Capítulo en el que se analizan los diferentes puntos de control a tener en cuenta en una base de datos, junto con un análisis pormenorizado de cómo obtener una base de datos segura y cómo gestionar las identidades con Oracle.

Puntos de control en una base de datos Oracle

Los sistemas de base de datos son elementos de la Tecnología de la Información (TI) esenciales, e importa su protección puesto que *"información es poder"*, y el dato y su acceso es el primer componente de ese poder. La importancia de estos activos es extrema, y el impacto ante su compromiso alto, por lo que los controles de seguridad han de reducir el factor de exposición al mínimo.

A continuación se expondrán una serie de puntos de control a tener en cuenta para mantener la seguridad en una base de datos.

Autenticación y autorización

Pese a que hoy en día la tecnología permite mecanismos de autenticación más avanzados, la contraseña sigue reinando, tanto en la autenticación de las conexiones desde las aplicaciones hacia la base de datos, como en la autenticación que exponen las propias aplicaciones a sus usuarios.

En la mayoría de las bases de datos, la comunicación por defecto entre la aplicación y la base de datos no usa cifrado, ni técnica alguna que evite interceptar esta comunicación. Además hay que tener en cuenta, que existe una complejidad inherente en el control de accesos en bases de datos, con múltiples usuarios con distintos permisos de acceso sobre multitud de objetos como tablas, vistas, procedimientos almacenados... Por esta razón resulta difícil aplicar correctamente el conocido principio de *mínimo privilegio* en la configuración de accesos.

En las bases de datos existen un gran número de privilegios (por ej. 173 en Oracle 10g), por ello es necesario disponer de personal capacitado para que asigne los correspondientes privilegios para los roles y cuentas.

Seguro tras la instalación

Tras la instalación, una base de datos es totalmente insegura por defecto. El principal agujero son los esquemas por defecto si no se toman precauciones elementales, como aplicar un proceso de configuración de seguridad, y resulta complejo seguir estos procesos sin que la base de datos deje de funcionar.

Las actuaciones obvias serían de distinta índole: Desde lo simple, como el cambio de contraseñas o el borrado de estos esquemas, a la revocación de derechos de conexión y/o roles asignados, bloqueo de las cuentas, expiración de sus contraseñas y auditoría de los intentos de uso, hasta configuraciones avanzadas como el uso de un *disparador* para evitar la autenticación con cuentas seleccionadas y alertar sobre su uso.

¿Por qué son creados los usuarios por defecto?

Las cuentas por defecto de Oracle pueden ser creadas por muchos motivos diferentes. Por ejemplo, las cuentas SYS y SYSTEM, DBSNMP y OUTLN a menudo son creadas por omisión cuando una base de datos es creada. Si la base de datos es creada usando el asistente el problema puede ser mucho más grande, ya que entre diez y veinte cuentas son creadas simplemente como parte de la creación de la base de datos.

Otras cuentas por defecto también pueden ser creadas después de la creación de la base de datos inicial, mediante la ejecución de scripts que residen en $ORACLE_HOME/rdbms/admin u otros directorios. Estos scripts suelen ser ejecutados para añadir un rasgo adicional o funcional o añadir el código de ejemplo de una base de datos.

¿Cuál es la cuestión?

Esta pregunta es a menudo pasada por alto por empresas que usan Oracle. En la mayoría de empresas los usuarios por defecto no son anulados o no se les ha cambiado la contraseña. Esto quiere decir que estas bases de datos han fallado en la medida de seguridad más simple. Se está permitiendo que exista una cuenta por la cual un atacante puede acceder a la base de datos y así conseguir información de la misma en el mejor de los casos, ya que a veces estas cuentas tienen privilegios de sistema con los que podría realizar cualquier cosa sobre la base de datos.

¿Está Oracle intentando remediar este problema?

Oracle ha hecho esfuerzos para reducir el problema, haciendo al instalador más inteligente y permitiendo a los clientes escoger si hay que instalar ejemplos. Aún así las cuentas por defecto todavía son instaladas por omisión. Oracle también ha hecho esfuerzos para cerrar y expirar las contraseñas sobre la mayor parte de las cuentas. Hoy en día no se permite escoger las contraseñas estándar para los usuarios SYS y SYSTEM, pero hay todavía un gran número de cuentas por defecto que pueden ser creadas en Oracle, cuyas contraseñas son sabidas y no son cerradas.

Este problema sigue creciendo debido a que con cada nueva versión de Oracle el número de cuentas por defecto posibles aumenta.

¿Qué se puede hacer?

La ayuda está al alcance de la mano. Las listas de contraseñas por defecto de Oracle se encuentran disponibles en muchas páginas web, con estas listas el usuario de Oracle puede asegurarse si su base de datos tiene alguna de estas cuentas habilitadas.

En caso de tener una cuenta por defecto habilitada puede realizar dos cosas. Primero debe decidir si la cuenta puede ser desactivada o eliminada, esto debería ser posible en casi todos los casos. En caso de no poder ser eliminada habría que cambiar la contraseña, cerrar y expirar la cuenta.

Auditabilidad

Aunque la activación de la auditoría en base de datos puede tener un impacto en el rendimiento. El objetivo no es activar la auditoría en su totalidad, sino auditar ciertas operaciones.

Como mínimo, habría que auditar los intentos de conexión, exitosos y fallidos, así como los accesos y alteraciones del propio registro de auditoría, o a las operaciones realizadas mediante cuentas privilegiadas.

A continuación se presentan una serie de cuestiones básicas que deben tenerse en cuenta al contemplar las características del uso de una auditoría en Oracle.

¿Por qué es necesaria la auditoría?

Muchas de las empresas no utilizan las características de la auditoría interna de Oracle. O bien, cuando lo hacen, quedan tan abrumados por la cantidad de elementos que se pueden auditar que no saben cuales auditar, de forma que conviertan todo en necesario, y luego se den cuenta que hay demasiada información para leer y *"digerir"* con rapidez en los datos generados por la auditoría.

La auditoría de Oracle puede ayudar a detectar el acceso no autorizado y el abuso interno de los datos, que se produce en la base de datos.

¿Cuándo deben ser auditados los usuarios de Oracle?

Sería conveniente tener un conjunto básico de acciones de auditoría activo todo el tiempo. El mínimo ideal es capturar el acceso de los usuarios, el uso de privilegios del sistema y los cambios en la estructura del esquema de la base de datos. Este conjunto básico no mostrará el intento de acceso a datos específicos que no deben ser accedidos, sin embargo, dará un panorama razonablemente simple del acceso *"incorrecto"* y el uso de privilegios.

Si un empleado es sospechoso de acciones inapropiadas o se ha sospechado un ataque, entonces puede activarse una auditoría más detallada para tablas específicas. Desde el punto de vista de gestión de datos, auditar los cambios de datos para todas las tablas no es realmente práctico y podría también afectar a su rendimiento.

¿Cómo pueden ser auditados los usuarios de Oracle?

Los comandos estándar de auditoría permitirán a todos los privilegios del sistema ser auditados, junto con el acceso a los objetos a nivel de cualquiera de las tablas o vistas en la base de datos para seleccionar, borrar, insertar o actualizar. La auditoría se puede ejecutar con éxito para intentos fructuosos o intentos infructuosos, o ambas cosas. Puede ser individual para cada usuario o para todos los usuarios.

¿Cuáles son los resultados y la complejidad?

Si en el proceso de auditoría están activadas muchas o todas las opciones, entonces la pista de auditoría resultante puede ser muy grande y difícil de interpretar y administrar. Por otra parte, si la auditoría se utiliza en todas las tablas y vistas de la base de datos, esto puede tener un efecto negativo sobre el rendimiento de la misma. Cada vez que una acción auditada se realiza, en la base de datos se escribe un registro auditable.

Para evitar que el resultado de la auditoría conlleve el posterior estudio de una gran volumetría de datos, hay que definir qué es lo que se quiere auditar. Para ello la clave es la sencillez y la prudencia. Debe utilizarse únicamente la auditoría que se necesita para ofrecer una visión global de lo que está sucediendo y realizar un seguimiento detallado de los objetos y datos críticos.

En caso de necesitar una auditoría más detallada, es importante definir qué acciones o abusos van a ser controlados de modo que pueda filtrarse el resultado de la auditoría para cada una de estas acciones.

Posteriormente, en este capítulo se profundizará en las características de la auditoría en Oracle.

Seguridad y gestión de identidades con Oracle

Una vez enumerados los diferentes puntos de control a tener en cuenta, se va a profundizar en los diferentes aspectos o características que proporciona Oracle.

Desde su fundación en 1977, Oracle se ha comprometido con la seguridad. Durante años, gobiernos y empresas de todo el mundo han llegado a confiar en Oracle por sus inigualables capacidades de seguridad. En general, la plataforma de seguridad de Oracle está compuesta por la *Base de Datos Oracle*, *Oracle Application Server* y *Oracle Identity Management*.

Oracle Identity Management permite centralizar la provisión y administración de usuarios de las aplicaciones, eliminando las molestias de mantenimientos asociados con la tradicional asociación de las aplicaciones una a una y combinaciones de nombre de usuario / contraseña.

A continuación se muestra una imagen con los componentes de Oracle Identity Management:

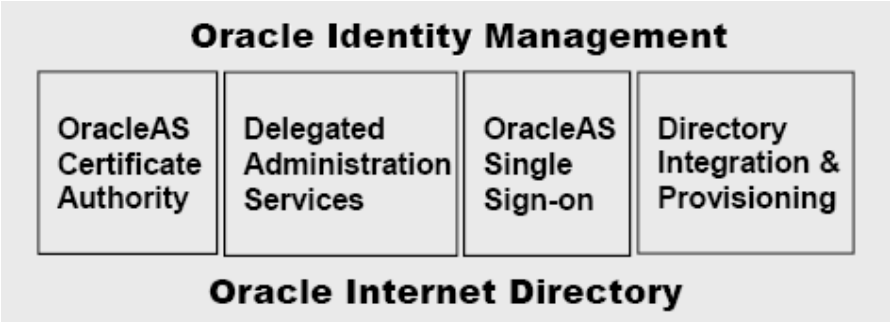

Imagen 4. Componentes de Oracle Identity Management

Fuente: Oracle Database 10g Security and Identity Management

- Oracle Internet Directory: Robusto LDAP V3 compatible con el directorio de servicios implementado en Oracle 9i.

- Directory Integration & Provisioning: Permite la sincronización entre Oracle Internet Directory y otros directorios y servicios de provisión automática de componentes de Oracle y las aplicaciones de terceros.

- Delegated Administration Services: Proporciona confianza proxy basada en la administración de la información del directorio por parte de los usuarios y administradores de la aplicación.

- Oracle Application Server Single Sign-On: Ofrece a los usuarios finales acceso a Oracle Single Sign-On y aplicaciones web de terceros.

- Oracle Application Server Certificate Authority: Gestiona y publica certificados X.509 V3 de apoyo a las tecnologías basadas en PKI (Infraestructura de clave pública), tales como autenticación, firma digital y S / MIME.

Seguridad del usuario

La gestión de la identidad es uno de los componentes operacionales más importante de TI en cualquier organización. La mayoría de las organizaciones se enfrentan a enormes obstáculos en la gestión de usuarios. Los usuarios a menudo dentro de una organización tienen demasiadas cuentas de usuario, un problema exacerbado por el crecimiento de las aplicaciones basadas en la web self-service. Las organizaciones quieren el acceso a los datos *"por usuario"* y no desean que la rendición de cuentas se convierta en una pesadilla administrativa de la gestión de los usuarios en cada base de datos en las que un usuario accede.

El problema se agrava por las aplicaciones web y e-business. Los socios y clientes no desean crear una cuenta de usuario para cada socio o cliente que accede a una de las bases de datos, sin embargo, los privilegios y la rendición de cuentas *"por socio"* es muy deseable. La seguridad de usuarios en Oracle, consiste tanto en la administración de privilegios como de esquemas, además de la gestión centralizada de la exigencia de los datos de acceso a cada uno de los usuarios.

Administración de Privilegios

Un desafío inherente a cualquier sistema distribuido, incluyendo los tres niveles de sistemas, es que la información común de las aplicaciones es a menudo fragmentada a través de la empresa, dando lugar a datos redundantes, inconsistentes, y caros de administrar.

Los directorios son el mejor mecanismo para que la información empresarial esté disponible para múltiples sistemas diferentes dentro de una empresa. Además, hacen posible el acceso a las organizaciones o compartir cierto tipo de información a través de Internet, por ejemplo, a través de una red virtual privada.

Un tipo específico de información que las empresas comúnmente proponen para el almacenamiento en un directorio son los privilegios y el control de acceso a la información. Los privilegios de usuario, representados como roles, limitaciones a objeto, representadas mediante listas de control de acceso (ACL) listan los usuarios que pueden tener acceso a un objeto, pueden ser almacenados en un directorio.

Los directorios de información que especifican los privilegios de usuario o atributos de acceso son sensibles, ya que la modificación no autorizada de esta información puede dar como resultado la concesión o denegación de privilegios o acceso a los usuarios de forma no autorizada. Un directorio debe mantener la información en nombre de la empresa, de manera que se pueda garantizar que sólo se autoriza a los administradores del sistema de seguridad el poder modificar los privilegios o acceso a la información mantenida en un directorio. *Oracle Internet Directory* soporta los atributos / niveles de control de acceso y opcionalmente la identificación de usuarios autenticados de forma segura a través de SSL, además puede ser configurado de forma que sólo usuarios específicos y que estén autenticados de forma segura se les permita actualizar la información del directorio acerca de los privilegios de usuario o el acceso.

Esquemas compartidos

El esquema independiente de usuario, o esquema compartido, extiende los beneficios de un directorio que permite la integración de la base de datos para delegar la administración de la identidad del usuario en el directorio, así como sus privilegios. La identidad de los usuarios se mantiene en un repositorio central LDAP, específicamente, Oracle Internet Directory. Cuando un usuario se conecta a través de un esquema independiente del usuario, el directorio determina si el usuario está registrado, y en caso afirmativo, en qué esquema de la base de datos debe estar asociado el usuario, y qué roles debe tener dicho usuario.

Por ejemplo, existen 500 usuarios de una aplicación y requieren el acceso a los datos en varios servidores de bases de datos en la empresa. En lugar de mantener 500 cuentas de usuario diferentes en cada base de datos, Oracle permite al administrador del sistema crear un único esquema compartido con los privilegios adecuados en cada base de datos y, a continuación, crear los 500 usuarios de la empresa en Oracle Internet Directory. Al conectarse a cualquier base de datos, los usuarios son asignados al esquemas apropiado sobre la base de datos y heredan los privilegios asociados al esquema, así como los privilegios adicionales que se asocian con los roles que se les conceden en el directorio. A pesar de que estos usuarios comparten un esquema común, la identidad de cada uno de los usuarios es individual.

La utilización del esquema compartido tiene una serie de beneficios. Se reduce la carga administrativa asociada con la gestión de los usuarios en una empresa, y permite una gestión eficaz mucho mayor de usuarios de lo que anteriormente era posible. Además, puede proporcionar un mecanismo para integrar la cuenta de usuario y gestión de privilegios a través de los múltiples niveles en un sistema multi-nivel, siempre y cuando se apoye en la gestión de identidades de usuario y privilegios en el directorio.

Password: Autenticación del grupo de usuarios

En Oracle 8i, la seguridad del grupo de usuarios se basó en cartera de clientes para autenticar al usuario. Esto requiere SSL para establecer canales seguros entre el cliente y el servidor, y la base de datos y un servidor LDAP en un directorio compatible. Este mecanismo de autenticación utiliza SSL y los certificados X.509 V3, los cuales requieren la instalación de paquetes Oracle tanto en el cliente como en el servidor.

Debido a que este sistema necesita un certificado X.509, para cada usuario de la empresa, expedido por una unidad certificadora de confianza, los gastos generales pueden ser importantes para las grandes empresas.

En Oracle, el grupo de usuarios puede utilizar password basados en autenticación, eliminando la exigencia de la cartera de clientes y la mayoría de Secure Sockect Layer (SSL) de transformación. Además los usuarios pueden usar un solo nombre de usuario y contraseña para conectarse a múltiples bases de datos, si así lo desea. Igualmente, Oracle provee al administrador la utilidad de migración de usuarios para migrar usuarios de múltiples e independientes bases de datos a un directorio de servicios central LDAP y así centralizar la gestión de usuarios y privilegios.

Oracle: Seguridad a nivel de fila

La seguridad a nivel de fila es la capacidad para controlar el acceso a las filas individuales dentro de una tabla de la base de datos después de que a un usuario de la aplicación se le haya dado el privilegio sobre esa tabla de la base de datos. Este tipo de control de acceso es difícil de implementar programando y normalmente aumenta la complejidad de la aplicación.

Oracle 8i estableció un nuevo estándar de seguridad en base de datos con la introducción de *Oracle Label Security* y la *base de datos virtual privada* (VPD – Virtual Private Database). En la versión 10g, Oracle introduce nuevas mejoras tanto en Oracle Label Security como a la base de datos virtual privada. Así, Oracle permite políticas Oracle Label Security que pueden ser administradas en la infraestructura Oracle Identity Management. Desde la versión 10g, la base de datos virtual privada de Oracle introduce la aplicación de políticas de seguridad sobre columnas y el enmascarado opcional de la columna.

Base de datos virtual privada

La base de datos virtual privada se introdujo en Oracle 8i e incluye seguridad a nivel de fila programable. Permite al desarrollador o DBA fijar la aplicación de una política de seguridad sobre una tabla, vista o sinónimo de la base de datos. La política de seguridad es invocada cuando una sentencia SQL hace referencia al objeto asociado con dicha política.

En una empresa, la utilización de una base de datos virtual privada puede desembocar en un menor costo en el despliegue de las aplicaciones. La seguridad puede ser construida una sola vez en la base de datos, en vez de en todas las aplicaciones que acceden a los datos. Proporciona una mayor seguridad porque es forzada por la base de datos, no importa como un usuario acceda a los datos.

El acceso directo o indirecto a una tabla con una política de seguridad asociada hace que la base de datos invoque la función que implementa dicha política. La función devolverá como resultado la condición de acceso, conocida como predicado (cláusula *where*) que la base de datos anexa a la sentencia SQL del usuario, por tanto modifica dinámicamente el acceso a los datos.

Por ejemplo, un sistema web de entrada puede hacer valer el acceso basado en el tipo de usuario, si el usuario es un cliente o un representante de ventas. De esta manera los clientes pueden ver el estado de sus pedidos, mientras que los representantes pueden ver las órdenes de pedido, pero solo las de sus clientes.

Base de datos virtual privada: Políticas de seguridad asociadas a columnas

Oracle permite a la base de datos virtual privada asociar políticas de seguridad a columnas de tablas de la aplicación. Sólo cuando se hace referencia a la columna de dicha tabla la política es invocada.

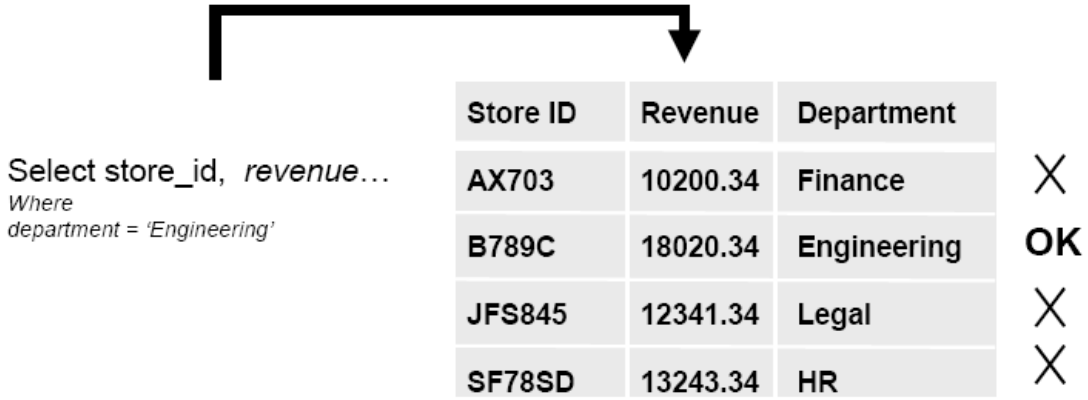

Tabla 4. Datos devueltos en la consulta con el uso de una política asociada a columnas

Fuente: Oracle Database 10g Security and Identity Management

En la consulta se hace referencia a la columna *"revenue"* (ingresos) la cual tiene asociada una política de seguridad. Cuando el departamento de ingeniería ejecuta la consulta sólo se mostrará el registro de su departamento.

Base de datos virtual privada: Políticas de seguridad asociadas a columnas y enmascarado del dato

Oracle introduce una nueva opción para hacer cumplir las políticas de seguridad asociadas a las columnas. Esta opción le dice a la base de datos para cada una de las filas de retorno, independientemente de la política de restricción, la máscara que debe mostrar para los valores de la columna para aquellas filas que no cumplan la política de seguridad. Por ejemplo, suponiendo que la política de seguridad de una base de datos virtual privada, es la política escrita en PL/SQL y asignada a la columna *"revenue"*. La sentencia SQL se ejecutará al recuperar todos los ingresos de todos los departamentos. En Oracle 8i y Oracle 9i únicamente las filas que concuerden con las del departamento de *"Engineering"* habrían sido devueltas. Desde la versión 10g, en Oracle, la política puede ser aplicada de tal manera que todas las filas sean devueltas, pero los valores de la columna *"revenue"* de aquellos departamentos que no sean *"Engineering"* aparecerán enmascarados.

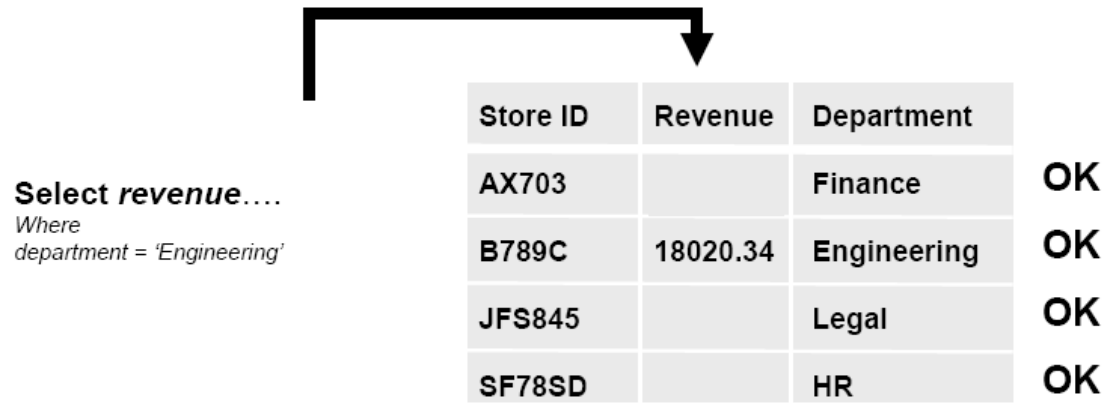

Tabla 5. Datos devueltos en la consulta con el uso de una política asociada a columnas con enmascarado del dato

Fuente: Oracle Database 10g Security and Identity Management

Oracle Label Security

Oracle Label Security se introdujo en Oracle 8i para sustituir a Trusted Oracle Multilevel Secure. A diferencia de la base de datos virtual privada donde los desarrolladores y DBA escriben la política de seguridad utilizando PL/SQL, Oracle Label Security ofrece un motor de seguridad y un diccionario de datos para la gestión del acceso a los datos usando etiquetas de sensibilidad. Una seguridad sofisticada a nivel de filas se puede lograr con poco o nada de programación requerida. Las *etiquetas de sensibilidad* son las que determinan la capacidad que tiene un usuario de ver o actualizar los datos de una aplicación. Igualmente, proporciona sofisticados controles que no son posibles con los privilegios a nivel de objetos.

Por ejemplo, suponiendo que una política de seguridad de una aplicación debe ser capaz de limitar el acceso a las órdenes de compra etiquetadas como *"Sensitive"*. Por defecto, un usuario con el privilegio *"Select"* sobre la aplicación podrá consultar toda la información de las órdenes de compra. Una solución a este problema sería crear dos vistas en la base de datos. La primera vista debe excluir los datos de las órdenes de compra etiquetadas como *"Sensitive"* y la segunda debe incluir todas las órdenes de compra. Este enfoque es problemático, porque la política de seguridad puede cambiar para incluir nuevos niveles de seguridad. Además, los usuarios de la aplicación tendrían que tener el rol correcto en función de su cargo, para que sólo pudiera ver la información relacionada con el mismo. Con las etiquetas de sensibilidad se ha resuelto este requisito de seguridad y, además, elimina la necesidad de crear vistas. Las etiquetas de sensibilidad pueden contener tres componentes: un único nivel jerárquico o clasificación, uno o más compartimentos o categorías y uno o más grupos.

Oracle Label Security proporciona una solución integrada para controlar el acceso a los datos basados en etiquetas de sensibilidad. La introducción de Oracle Label Security en una aplicación existente tiene las siguientes ventajas:

- Simplifica la aplicación.
- Aumenta la seguridad por el control de acceso en la base de datos.
- Crea una ventaja competitiva sobre las aplicaciones

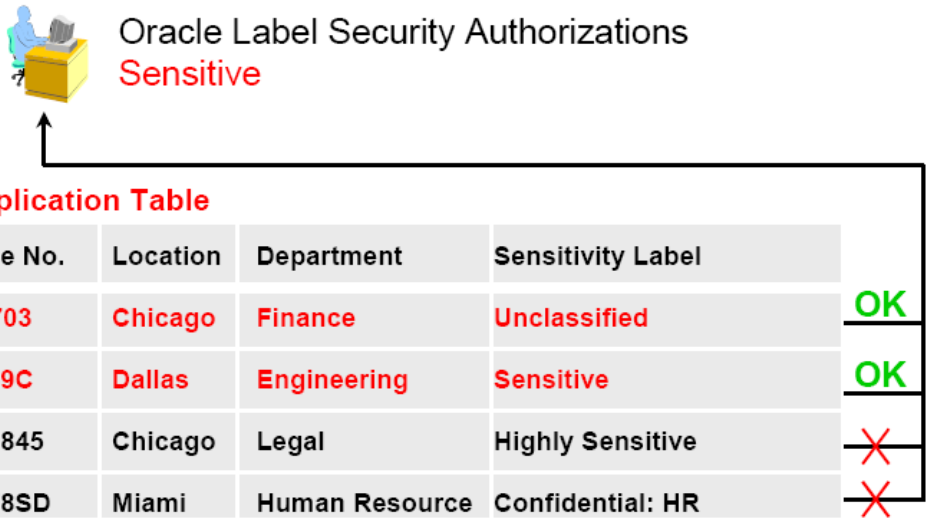

Tabla 6. Datos devueltos en la consulta con la utilización de Oracle Label Security

Fuente: Oracle Database 10g Security and Identity Management

En la imagen se muestra, como un usuario tiene asignada la autorización de etiqueta *"Sensitive"*, por tanto se le mostrarán las filas de la tabla que tengan la etiqueta de sensibilidad *"Sensitive"* o de menor sensibilidad.

Una aproximación a las etiquetas de sensibilidad

Las etiquetas de sensibilidad son fundamentales para Oracle Label Security. Estas son las que determinan si los usuarios de una aplicación tienen la capacidad de ver y actualizar los datos de una aplicación. Las etiquetas de sensibilidad proporcionan sofisticados controles que no son posibles con los tradicionales privilegios a nivel de objetos. Las etiquetas de sensibilidad pueden contener tres componentes: un único nivel jerárquico, uno o más compartimentos o categorías y uno o más grupos.

Nivel - El nivel jerárquico es el componente que denota la sensibilidad de los datos.

Compartimiento - El compartimiento es a veces denominado como categoría y no es jerárquico. Normalmente uno o más compartimientos se definen para segregar los datos. Por ejemplo, un compartimiento puede ser definido para una iniciativa estratégica en curso o asociar los usuarios subscritos a una aplicación.

Grupo - El grupo se utiliza para registrar la propiedad y se puede utilizar jerárquicamente.

Representación

La representación exterior de una etiqueta se compone de tres componentes, separados por dos puntos. La etiqueta *"Confidencial: Adquisiciones: Asía"* está integrado por los tres componentes de etiqueta siguientes:

Level = Confidencial

Compartment = Adquisiciones

Group = Asía

Una etiqueta puede tener un máximo de 4000 caracteres de largo y potencialmente se compone de decenas de compartimentos y grupos.

Múltiples políticas en etiquetas de seguridad

Oracle Label Security soporta múltiples políticas en una única base de datos. Una política es simplemente un identificador o el nombre asignado a un grupo de etiquetas de sensibilidad, autorización de etiquetas de usuario o autorizaciones de seguridad y privilegios de acceso de usuario.

Una sola base de datos puede tener múltiples políticas de control de acceso a datos diferentes.

- Etiquetas de seguridad: Política de privacidad

Level – Nivel	Compartments - Compartimentos	Groups - Grupos
Confidential Sensitive Highly Sensitive	Personally Identifiable Information	HR Senior Management

Tabla 7. Etiquetas de seguridad: Política de privacidad

- Etiquetas de seguridad: Política de ingeniería

Level – Nivel	Compartments - Compartimentos	Groups - Grupos
Confidential Sensitive		IT Security Security Development General Development

Tabla 8. Etiquetas de seguridad: Política de ingeniería

Los datos pueden ser etiquetados con cualquier combinación de los niveles, los compartimientos y los grupos especificados para una política individual.

Label 1 = Confidential

Label 2 = Sensitive : HR

Label 3 = Sensitive : HR : VP_GRP

Label 4 = Highly_Sensitive

Label n = Level : Compartments : Groups

- Autorización de etiquetas de usuario

Después de que los componentes individuales de la etiqueta se hayan creado, a los usuarios de la aplicación se les puede asignar las autorizaciones de etiqueta. Para cada una de las políticas, un usuario puede disponer de lo siguiente:

- Máximo y mínimo nivel de sensibilidad.
- Cero o más compartimientos.
- Cero o más grupos

Para cada compartimiento o grupo un usuario puede acceso de lectura o de lectura/escritura.

- Programas de confianza almacenados en unidades

Oracle Label Security apoya la confianza en programas almacenados en unidades. Los procedimientos almacenados se les pueden asignar privilegios especiales que permiten al procedimiento almacenado eludir la etiqueta de seguridad encargada de hacer cumplir los controles. Esta funcionalidad es útil cuando un procedimiento almacenado se utiliza para la presentación de informes especiales o cálculos.

- Predicados SQL

Las políticas de Oracle Label Security pueden extenderse mediante la adición de predicados SQL a la aplicación de la política. Los predicados SQL se utilizan para proporcionar extensibilidad a la aplicación selectiva de normas de acceso a datos. Oracle Label Security proporciona una interfaz para añadir fácilmente predicados SQL a las políticas de Oracle Label Security. Por ejemplo, los siguientes predicados SQL podrían añadirse:

Ejemplo 1: *AND myfunction(col1) = 1*

Ejemplo 2: *OR SYS_CONTEXT('USERENV', 'SESSION_USSER') = name*

- Etiquetas de seguridad: Mediación de acceso

Oracle Label Security media el acceso mediante la comparación de las etiquetas de sensibilidad con la autorización de etiqueta asignada al usuario.

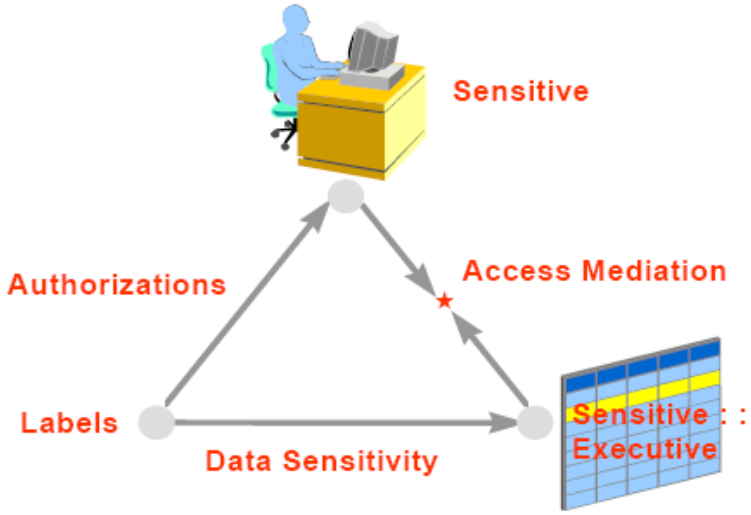

Imagen 5. Gráfico de mediación de acceso de Oracle Label Security

Fuente: Oracle Database 10g Security and Identity Management

Oracle: Administrador de políticas

Oracle Policy Manager es una herramienta de administración tanto de Oracle Label Security como de Base de datos virtual privada de Oracle. Las políticas de Oracle Label Security pueden ser gestionadas usando la herramienta Oracle Policy Manager mediante la conexión como el usuario *LBACSYS* u otro usuario con privilegios adecuados.

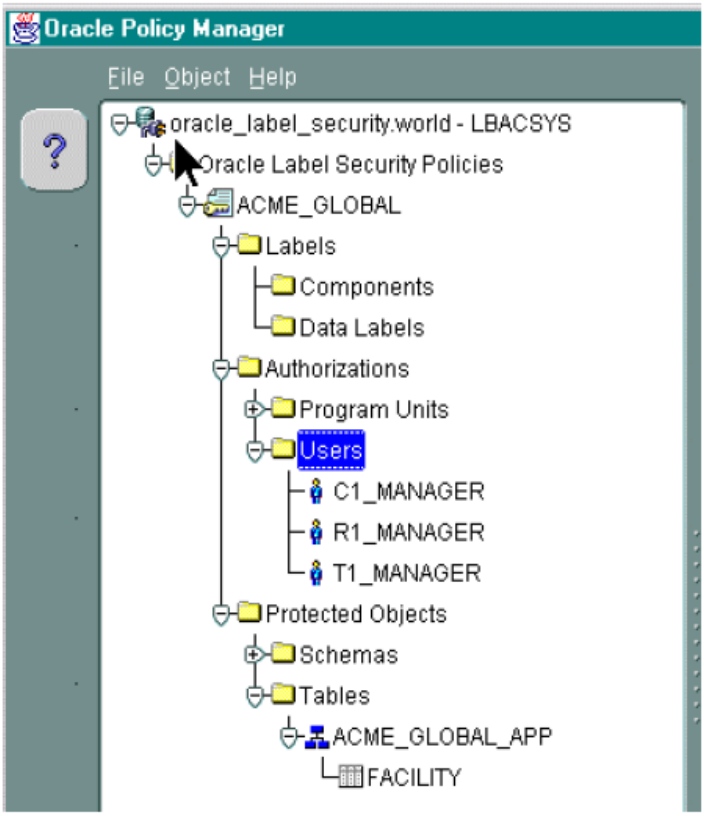

Imagen 6. Oracle Policy Manager

Fuente: Oracle Database 10g Security and Identity Management

Base de datos virtual privada y Label Security

Para decidir que tecnología debe utilizarse, inicialmente se debe entender los problemas de negocio asociados a la aplicación.

Debido a sus características, Oracle Label Security está diseñado para hacer frente a los siguientes problemas:

- Ejecución de la seguridad a nivel de fila basado en la sensibilidad de los datos, su categoría y/o las propiedades de la organización.

- Aplicación de requisitos de seguridad multinivel.

- Implementación sofisticada de los datos entre organizaciones anfitrión y organismos.

Mientras tanto, con la base de datos virtual privada se puede hacer frente a la seguridad a nivel de fila programable como solución, permitiendo ser diseñada para satisfacer una serie de necesidades específicas.

Para decidir que tecnología se ajusta mejor a las necesidades de una aplicación, habría que estudiar el tamaño de la base de datos, volumen de datos, número de clientes... Normalmente es aconsejable utilizar solamente la base de datos virtual privada para aplicaciones pequeñas o que por sus características no vayan a variar mucho sus necesidades con el paso del tiempo.

Cifrado de datos selectivos

Entre otras tecnologías de seguridad, Oracle protege los datos a través de fuertes sistemas e-business, basados en estándares de cifrado. Oracle ha apoyado el cifrado de los datos de la red a través de *Oracle Avanced Security* desde Oracle 7. Oracle también apoya la protección de determinados datos a través del cifrado dentro de la base de datos. Aunque el cifrado no es un sustituto eficaz para el control de acceso, uno puede obtener una medida de seguridad adicional por cifrado de manera selectiva de los datos sensibles antes de que se almacenen en la base de datos. Por ejemplo:

- Números de una tarjeta de crédito.

- Documento Nacional de Identidad.

Para hacer frente a la necesidad de cifrado de datos selectiva, Oracle ofrece un paquete PL/SQL para cifrar y descifrar los datos almacenados. El paquete se llama *DBMS_OBFUSCATION_TOOLKIT*, fue introducido en Oracle 8i y apoya el cifrado de la mayor parte de los datos usando:

Cifrado de datos estándar (DES) y (3DES).

Oracle: Cifrado de datos

Desde la versión 10g, Oracle introduce un nuevo paquete de herramientas llamado *DBMS_CRYPTO*.

Esta herramienta apoya la funcionalidad proporcionada por *DBMS_OBFUSCATION_TOOLKIT* y es más fácil de usar. También se incluye soporte para el cifrado de datos utilizando *Avanced Encryption Standard* (AES) (Estándar de cifrado avanzado). Además, el nuevo juego de herramientas proporciona apoyo para el cifrado de tipos de datos adicionales.

El conjunto de herramientas apoya la seguridad criptográfica de MD5 para garantizar la integridad de los datos.

Auditoría

Un aspecto crítico de cualquier política de seguridad es mantener un registro de la actividad del sistema para garantizar que los usuarios son responsables de sus acciones. La auditoría ayuda a disuadir el comportamiento de los usuarios no autorizados que no podrá ser impedido de otra manera. Es especialmente útil para garantizar que los usuarios autorizados del sistema no abusen de sus privilegios. Oracle se basa en las actuales, sólidas y amplias capacidades de auditoría de base de datos a fin de incluir el *fine-grained* (grano fino) en la auditoría, que puede servir como un *"sistema de alerta temprana"* de los usuarios que abusan de los privilegios de acceso a datos, así como un sistema de detección de intrusos para la propia base de datos.

Auditoría robusta y comprensible

El servicio de auditoría de Oracle permite a las empresas auditar la actividad de la base de datos por estados, mediante el uso del sistema de privilegios por objeto o por usuario. Por ejemplo, se puede auditar la actividad general, como todas las conexiones de usuarios a la base de datos, o algo más específico como la creación de una tabla por un usuario en particular. Se pueden auditar solamente las operaciones realizadas con éxito o las que no se han realizado con éxito. Por ejemplo, se podrá auditar las *"SELECT"* que realizan los usuarios sobre alguna tabla de datos sobre la que no tienen privilegios para ver su contenido.

Los registros de la pista de auditoría se pueden almacenar en una tabla de Oracle, haciendo que la información esté disponible para su visualización a través de consultas hoc o cualquier aplicación o herramienta, o en combinación del sistema operativo de auditoría sobre determinados sistemas operativos, para facilitar la gestión.

Auditoría eficiente

Oracle lleva a cabo la auditoría de manera eficiente: las declaraciones son analizadas conjuntamente una vez tanto para la ejecución como para la auditoría, nunca por separado. Asimismo, la auditoría se llevará a cabo en el propio servidor, con el objetivo de no sobrecargar la red. La granularidad y el alcance de estas opciones de auditoría de Oracle permiten a los clientes registrar y supervisar la actividad específica de bases de datos sin incurrir en el desempeño que conlleva la auditoría general.

Auditoría personalizada

Para grabar información personalizada que no está automáticamente incluida en registros de auditoría, Oracle puede utilizar disparadores (triggers) para personalizar aún más las condiciones de auditoría y la auditoría del contenido de registros. Los disparadores de la base de datos son definidos por el usuario mediante declaraciones PL/SQL o Java, almacenados de forma compilada. Mientras los usuarios ejecutan explícitamente procedimientos almacenados, los disparadores de la base de datos se ejecutan automáticamente en el servidor de datos basado en eventos pre-especificados. Un disparador es definido para que se ejecute antes o después de un *insert*, *update* o *delete*, de manera que cuando esa operación se realiza sobre la tabla en cuestión, el disparador se lanza automáticamente. Por ejemplo, se podría definir un disparador sobre la tabla *"Empleado"* para generar un registro de auditoría cada vez que un sueldo de empleado se incrementa en más de un 10 por ciento e incluir la información seleccionada, como era antes y después los valores de su salario.

Fine-grained aplicado a la auditoría

Oracle amplia la robustez, las capacidades de auditoría granular de la base de datos mediante la introducción extensible de la auditoría de grano fino (fine-grained). La auditoría de grano fino permite a las organizaciones definir políticas específicas de auditoría que pueden alertar a los administradores del uso indebido de los derechos de acceso legítimos.

La auditoría de grano fino permite a las organizaciones definir políticas de auditoría, que se especifican en las condiciones de acceso a datos que activan el evento de auditoría, y un uso flexible de eventos para notificar a los administradores que el hecho determinante ha ocurrido. Por ejemplo, una organización puede permitir que los rrhh pueda acceder a la información de los sueldos de los empleados, pero auditando el acceso cuando se accede a los salarios superiores a 90.000€. La política de auditoría (cuando sueldo > 90.000) se aplica a la tabla de empleados a través de una interfaz de auditoría (un paquete PL/SQL). La auditoría de grano fino de Oracle se extiende a las declaraciones *insert*, *update* y *delete*.

La auditoría de grano fino permite a las organizaciones perfeccionar sus capacidades de auditoría, para capturar e identificar particularmente el acceso a los datos específicos de preocupación. Además de proporcionar más granularidad, información específica de auditoría, tales como la detección del uso indebido de acceso legítimo, la auditoría de grano fino también puede servir como un mecanismo de detección de intrusos para la base de datos Oracle en sí misma.

Capítulo 5: Proceso de instalación de Oracle 11g

En este apartado se explicará cómo instalar y configurar Oracle 11g junto con Oracle Label Security. Para ello se mostrarán las imágenes con los pasos a seguir.

Al ejecutar el fichero "setup.exe" para iniciar el proceso de instalación, aparecerá la pantalla inicial.

Imagen 7. Proceso de instalación de Oracle 11g: Selección del método de instalación

En esta pantalla se podrá seleccionar el tipo de instalación que se desea realizar. En nuestro caso, al desear instalar Oracle Label Security, se pulsará "Instalación Avanzada".

Al pulsar el botón “Siguiente”, aparecerá la pantalla en la que se debe seleccionar el tipo de instalación que desea realizarse.

Imagen 8. Proceso de instalación de Oracle 11g: Selección del tipo de instalación

A continuación se describen los diferentes tipos de instalación que pueden realizarse.

Tipos de instalación:

- *Enterprise Edition.* Este tipo de instalación está diseñado para aplicaciones a nivel de empresa. Es la primera base de datos diseñada para cuadrícula, es una base de datos de gestión automática que tiene las funciones de escalabilidad, rendimiento, alta disponibilidad y seguridad necesarias para ejecutar las aplicaciones críticas más exigentes.

- *Standard Edition.* Este tipo de instalación está diseñado para aplicaciones a nivel de departamento o grupo de trabajo o para pequeñas y medianas empresas. Está diseñado para proporcionar las opciones y servicios de gestión de bases de datos relacionales esenciales.

- *Personal Edition.* Este tipo de instalación instala el mismo software que el tipo de instalación *Enterprise Edition*, pero sólo soporta un entorno de desarrollo y despliegue monousuario.

- *Personalizada.* Permite seleccionar los componentes concretos que se desean instalar. En nuestro caso, ésta será la opción marcada.

Posteriormente, se indicará la ubicación donde almacenar el software de la base de datos.

Imagen 9. Proceso de instalación de Oracle 11g: Ubicación de instalación

El asistente de instalación verificará si el entorno cumple todos los requisitos mínimos para instalar y configurar los productos seleccionados. Si hay algún elemento marcado con advertencia se deberá comprobar manualmente.

Imagen 10. Proceso de instalación de Oracle 11g: Comprobación de requisitos específicos del producto

Al pulsar el botón “Siguiente”, en caso de que todo se haya desarrollado de forma satisfactoria, aparecerá una ventana en la que se indicará que productos de Oracle 11g se desean instalar.

Imagen 11. Proceso de instalación de Oracle 11g: Selección de componentes para la instalación

Como se muestra en la imagen, se ha marcado la opción “Oracle Label Security 11.1.0.6.0” junto con el resto de opciones que vienen marcadas por defecto.

A continuación, en la pantalla de "Creación de Base de Datos" se podrá seleccionar entre una de las siguientes opciones:

- Crear Base de datos
- Configurar Gestión Automática de Almacenamiento (ASM)
- Instalar sólo Software de Base de Datos

Se marcará la opción "Crear Base de Datos", así una vez se haya instalado el software de Oracle, se ejecutará automáticamente el asistente para crear y configurar una base de datos.

Imagen 12. Proceso de instalación de Oracle 11g: Creación de Base de Datos

Finalmente se mostrará la ventana resumen, en la que se muestran los productos de Oracle 11g a ser instalados.

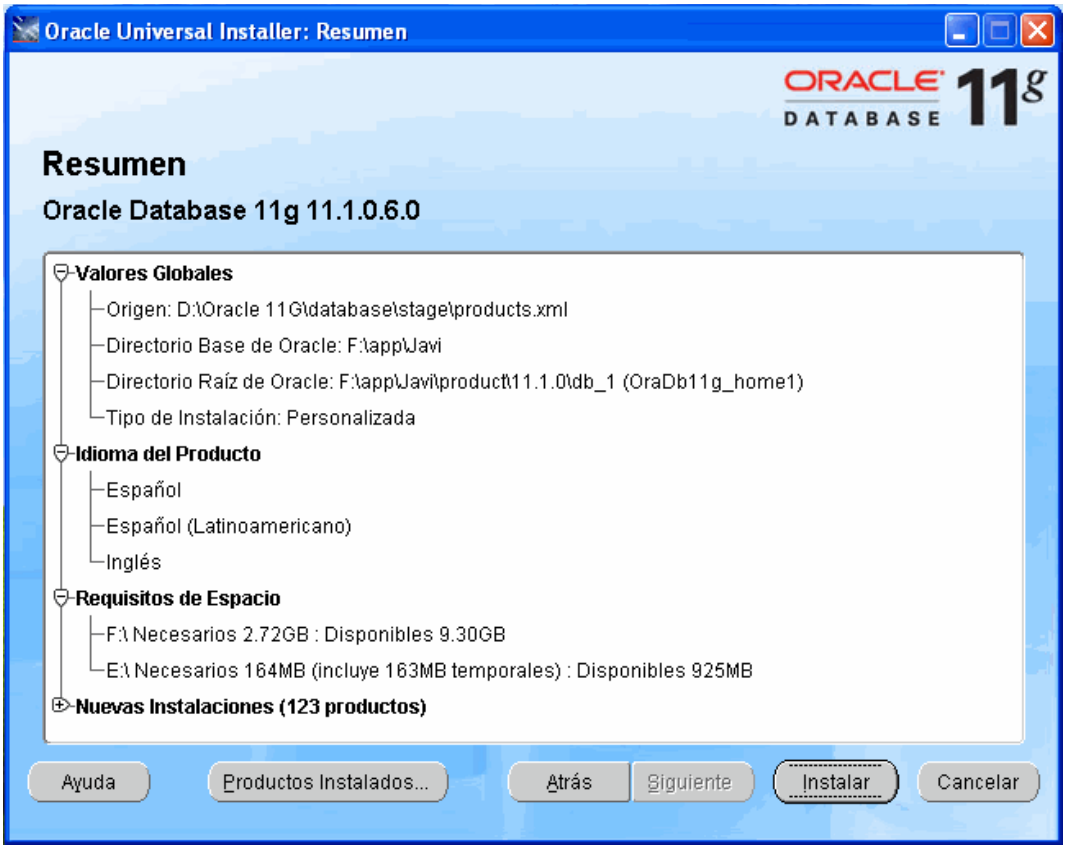

Imagen 13. Proceso de instalación de Oracle 11g: Resumen de productos a instalar

Al pulsar el botón “Instalar”, se iniciará el proceso de instalación del software de la base de datos.

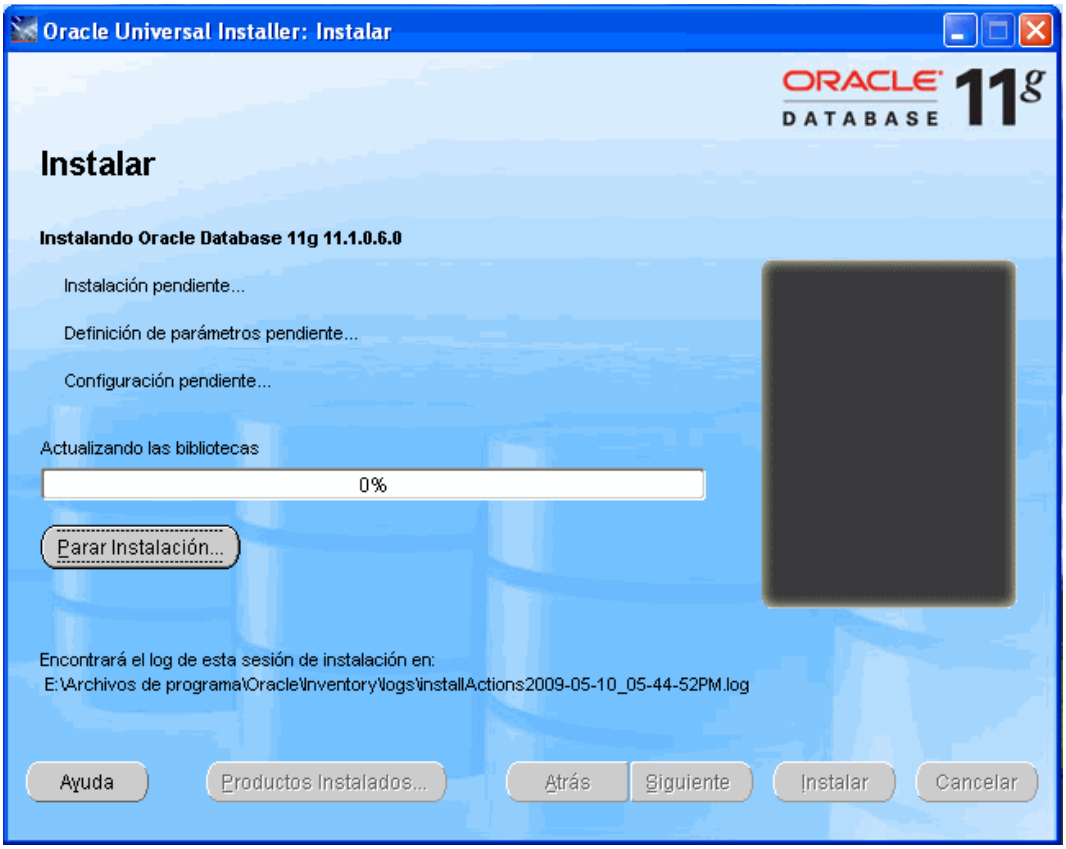

Imagen 14. Proceso de instalación de Oracle 11g: Evolución del proceso de instalación

Una vez haya finalizado el proceso de instalación del software. Se ejecutará automáticamente el asistente de configuración de red de Oracle.

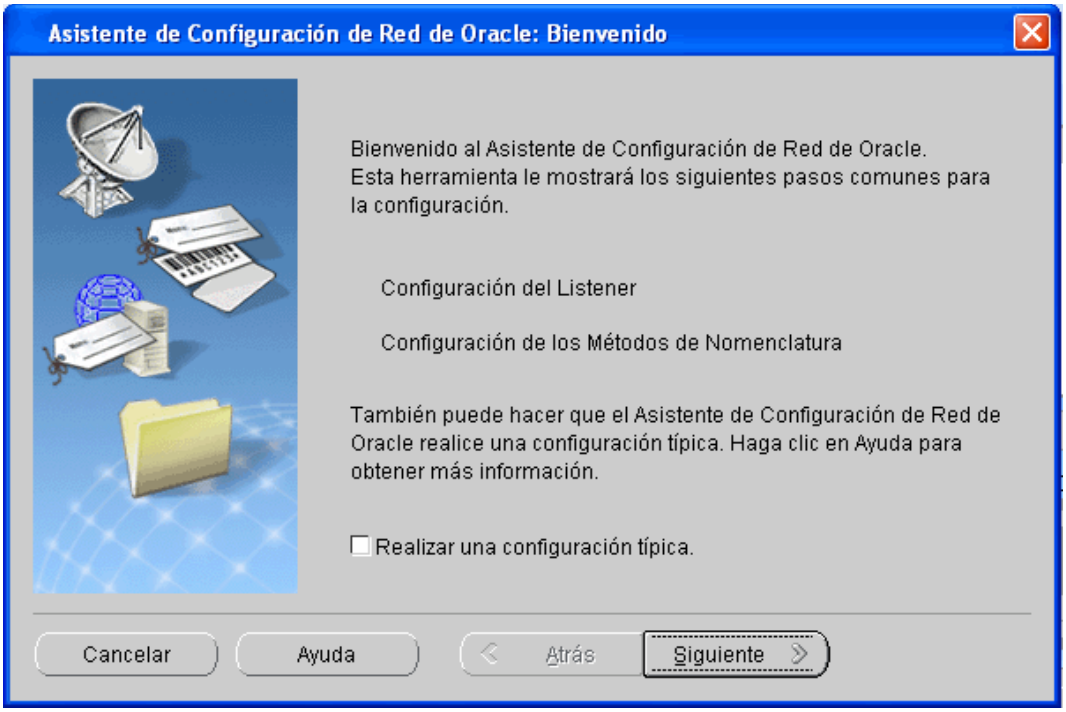

Imagen 15. Proceso de instalación de Oracle 11g: Asistente de configuración de Red de Oracle

Pulsando el botón “Siguiente”, aparecerá la siguiente pantalla en la que se indicará el nombre del listener.

Imagen 16. Proceso de instalación de Oracle 11g: Nombre del listener

Una vez indicado el nombre del mismo, se pulsará el botón “Siguiente”. A continuación, se especificará que protocolos se desean configurar en el listener.

Imagen 17. Proceso de instalación de Oracle 11g: Selección del protocolo del listener

Al pulsar el botón “Siguiente”, aparecerá la siguiente pantalla, en la que se deberá indicar el número de puerto TCP/IP a utilizar por el listener.

Imagen 18. Proceso de instalación de Oracle 11g: Número de puerto TCP/IP a utilizar por el listener

En este caso, se dejará seleccionada la opción por defecto y se pulsará al botón “Siguiente”.

A continuación, aparecerá la pantalla en la que se da la opción de configurar otro listener.

Imagen 19. Proceso de instalación de Oracle 11g: ¿Configuración de otro listener?

Al no querer configurar otro listener adicional, se dejará marcado “No” y se pulsará el botón “Siguiente”.

Ahora, el asistente nos mostrará una pantalla en la que se pregunta si se desea configurar algún método de nomenclatura adicional.

Imagen 20. Proceso de instalación de Oracle 11g: ¿Configuración de algún método de nomenclatura adicional?

En nuestro caso, se seleccionará la opción “No, no deseo configurar métodos de nomenclatura adicionales” y se pulsará “Siguiente”.

Finalmente, aparecerá la pantalla de finalización de configuración de la red de Oracle.

Imagen 21. Proceso de instalación de Oracle 11g: Finalización del asistente de configuración de Red de Oracle

Al pulsar el botón “Terminar” este asistente se cerrará y se ejecutará automáticamente el asistente de configuración de la base de datos.

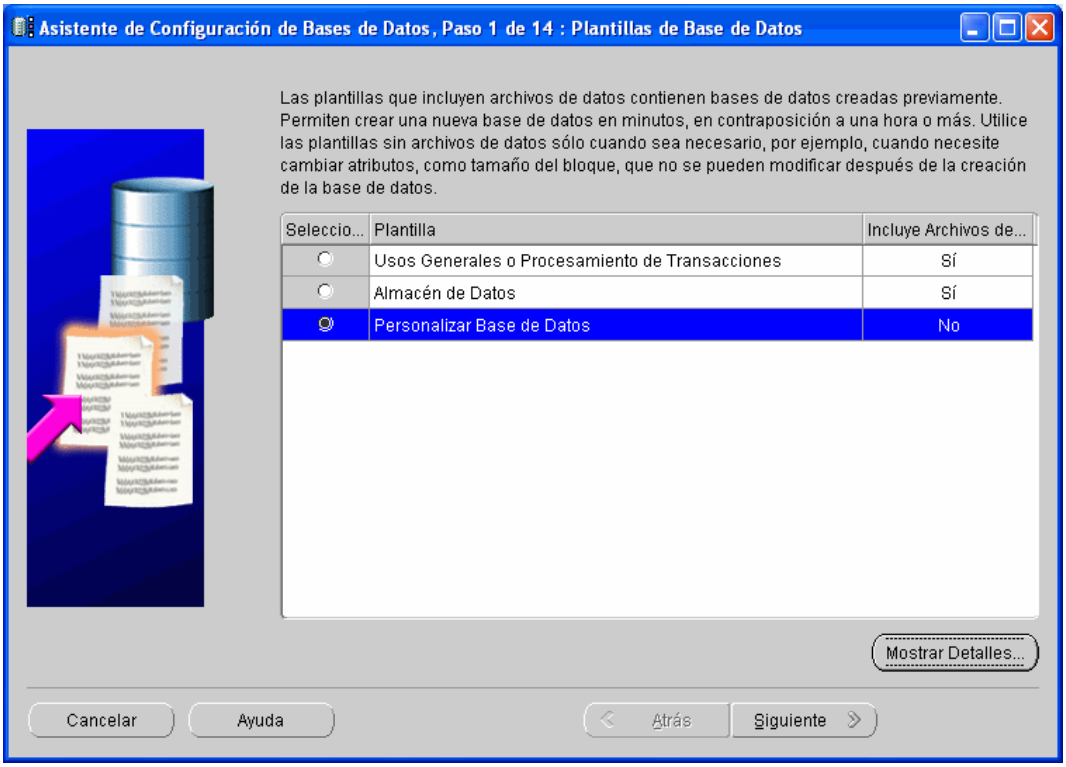

Imagen 22. Proceso de instalación de Oracle 11g: Asistente de Configuración de Bases de Datos

En esta primera pantalla del asistente (Paso 1), se deberá seleccionar la opción “Personalizar Base de Datos” y pulsar el botón “Siguiente”.

A continuación (Paso 2), debe especificarse el nombre de la base de datos y el SID (Identificador del Sistema Oracle).

Imagen 23. Proceso de instalación de Oracle 11g: Identificación de Base de Datos

Una vez se hayan indicado, se deberá pulsar el botón “Siguiente”.

En el paso 3, deberán seleccionarse las opciones de gestión. Se dejarán las opciones marcadas por defecto y se pulsará el botón “Siguiente”.

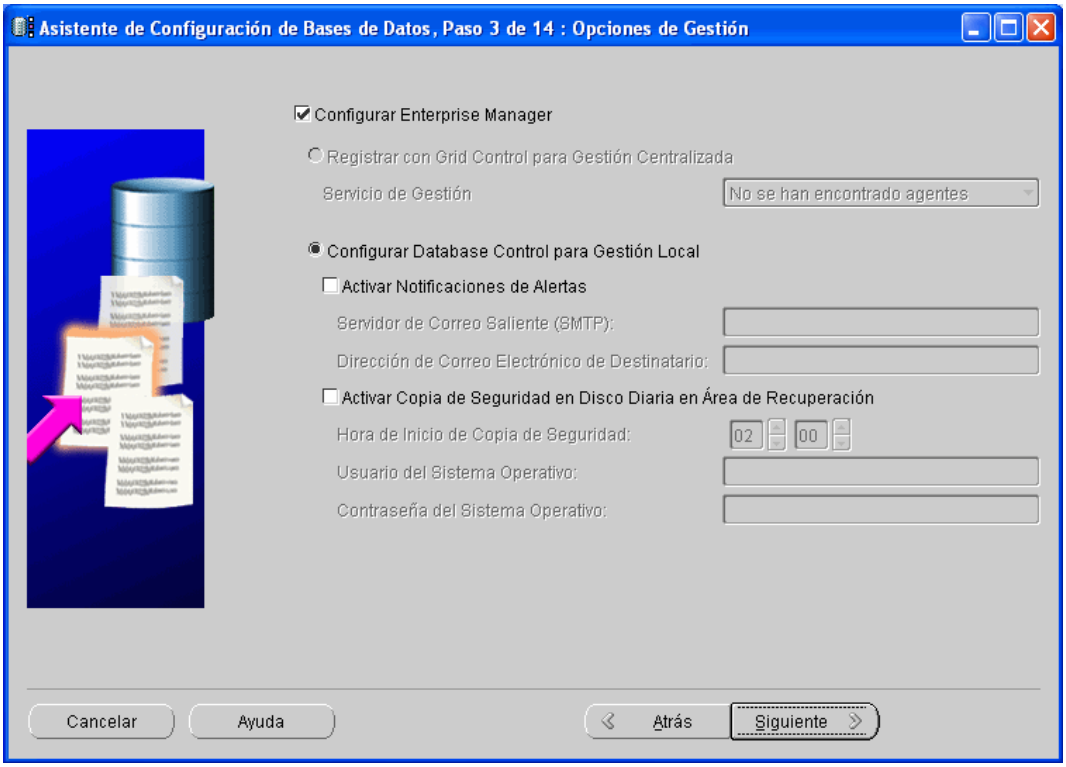

Imagen 24. Proceso de instalación de Oracle 11g: Opciones de gestión

A continuación, se podrá especificar las contraseñas de los diferentes usuarios administrativos, o usar una misma contraseña para todas las cuentas.

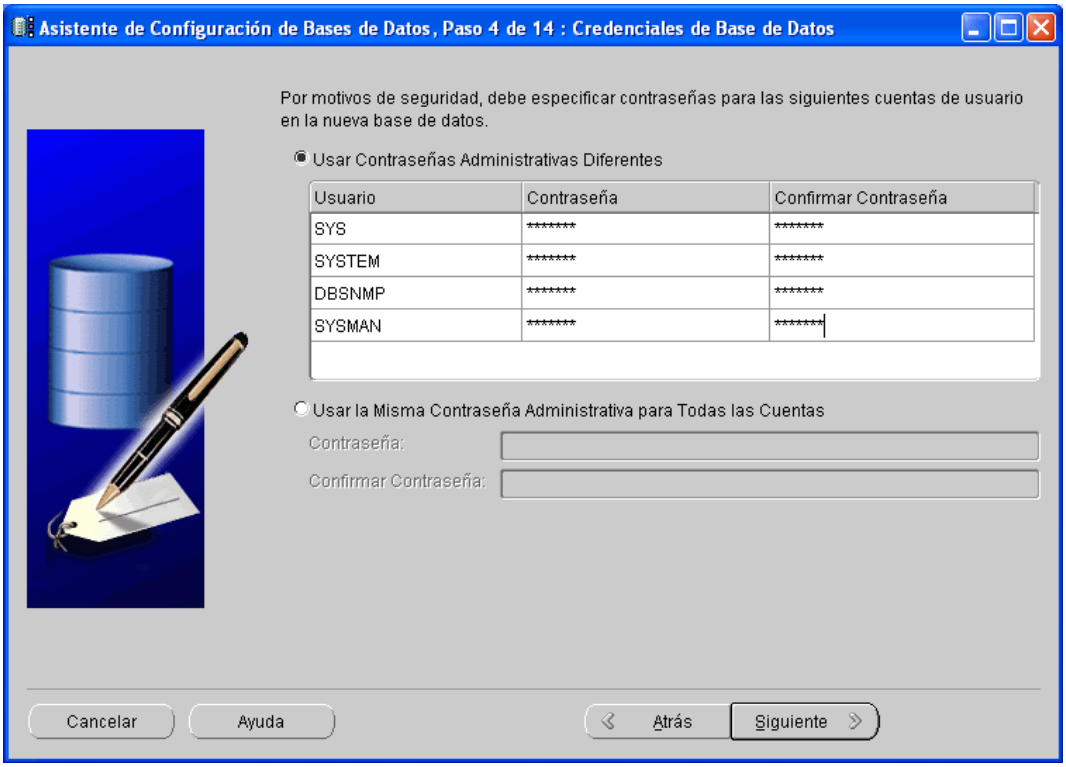

Imagen 25. Proceso de instalación de Oracle 11g: Credenciales de Base de Datos

Por seguridad, se indicará una contraseña diferente para cada uno de los usuarios. Posteriormente, se pulsará el botón “Siguiente”.

En los pasos que se muestran a continuación (hasta el paso 11), se ha dejado seleccionada la opción que el asistente marca por defecto y se ha pulsado el botón "Siguiente".

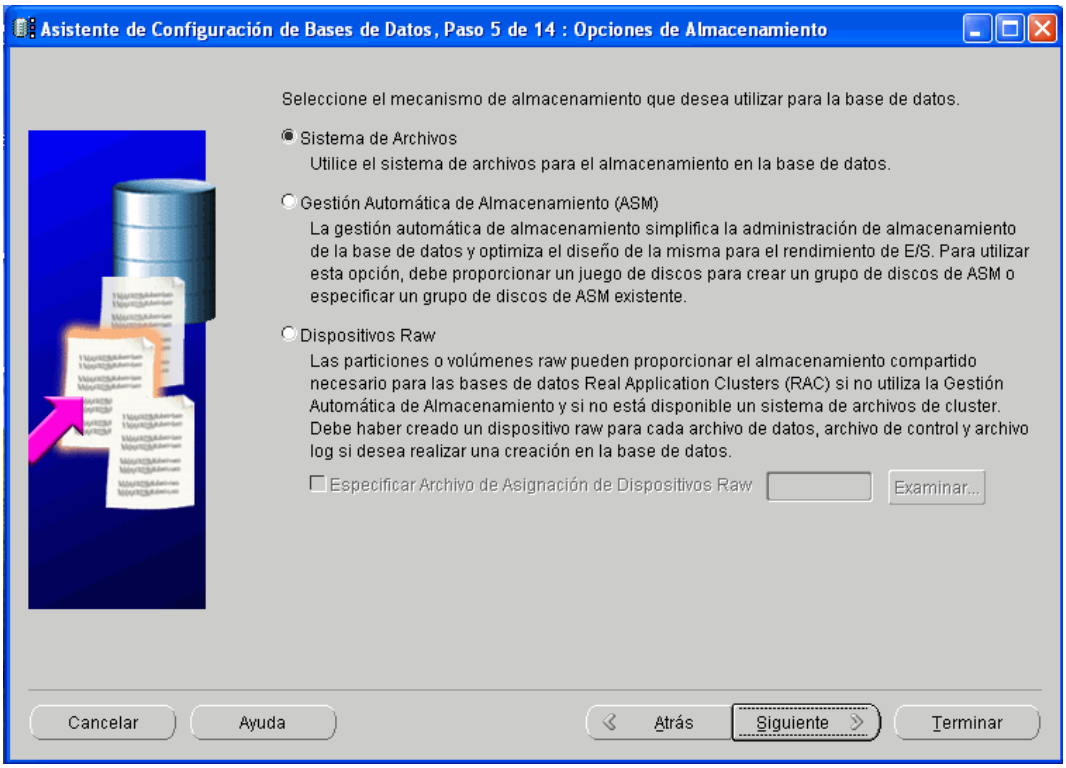

Imagen 26. Proceso de instalación de Oracle 11g: Opciones de almacenamiento

Imagen 27. Proceso de instalación de Oracle 11g: Ubicaciones de archivos de Base de Datos

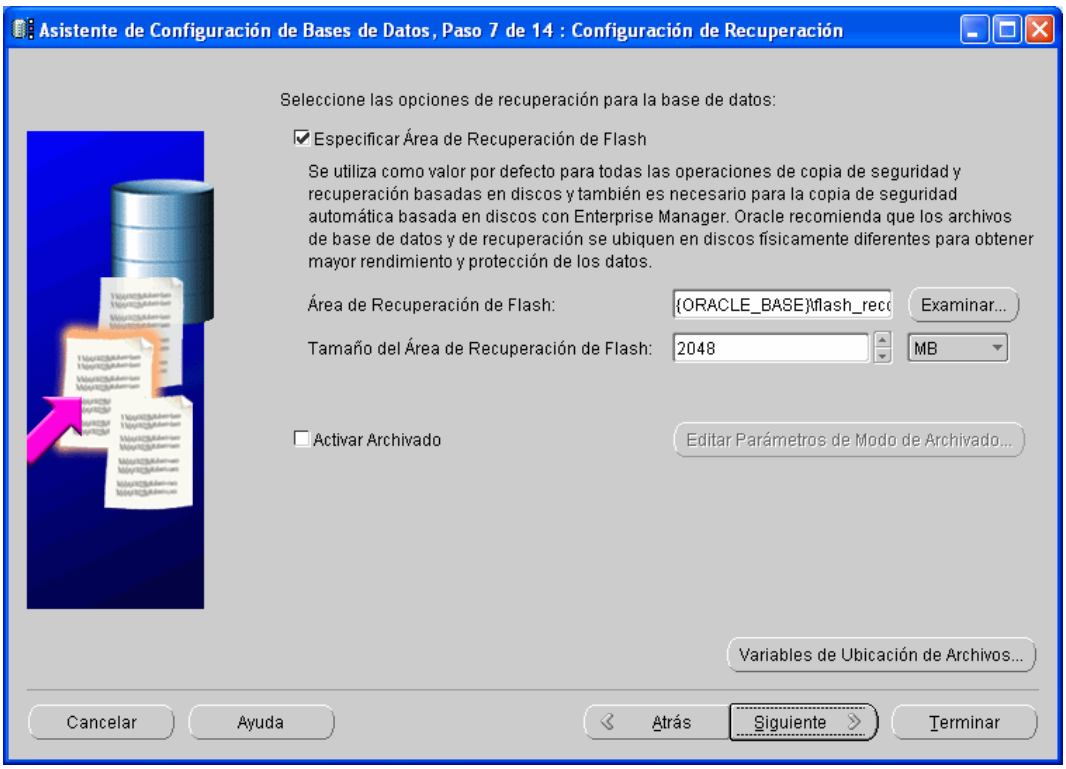

Imagen 28. Proceso de instalación de Oracle 11g: Configuración de recuperación

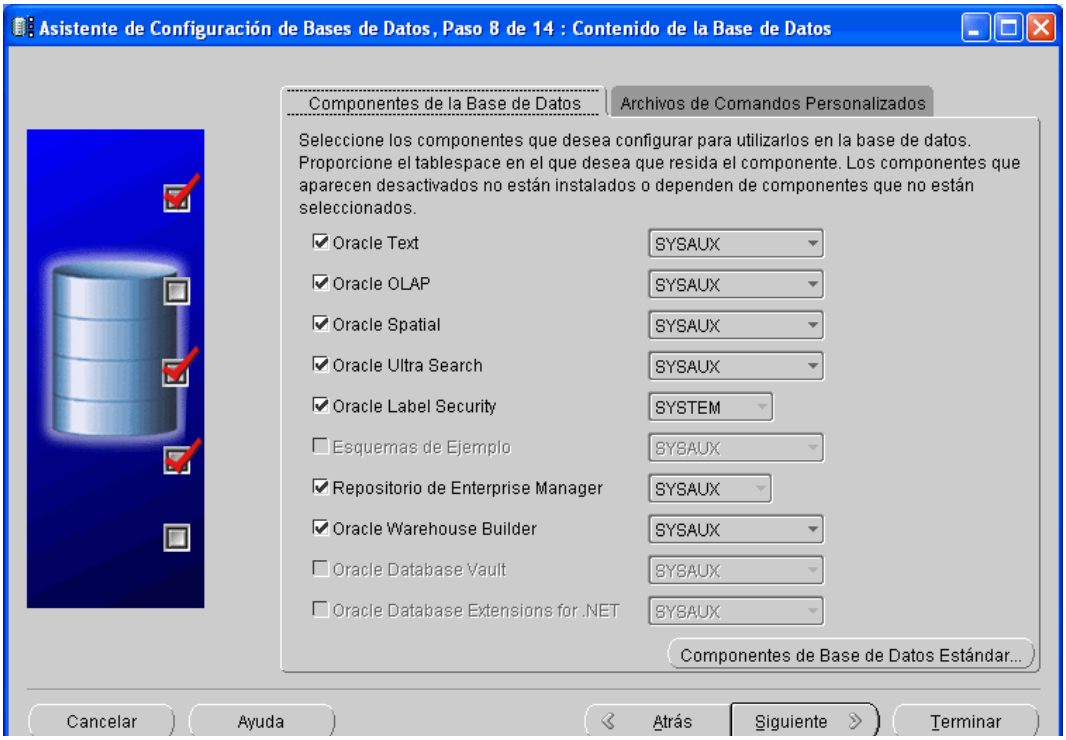

Imagen 29. Proceso de instalación de Oracle 11g: Contenido de la Base de Datos

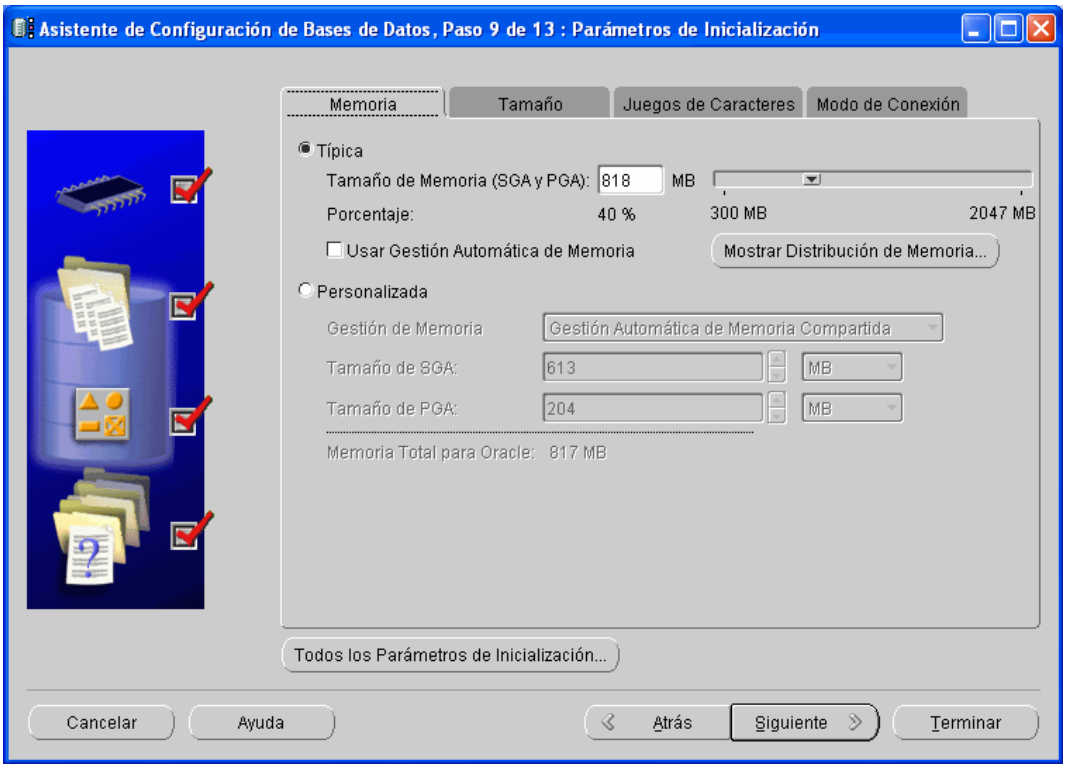

Imagen 30. Proceso de instalación de Oracle 11g: Parámetros de Inicialización

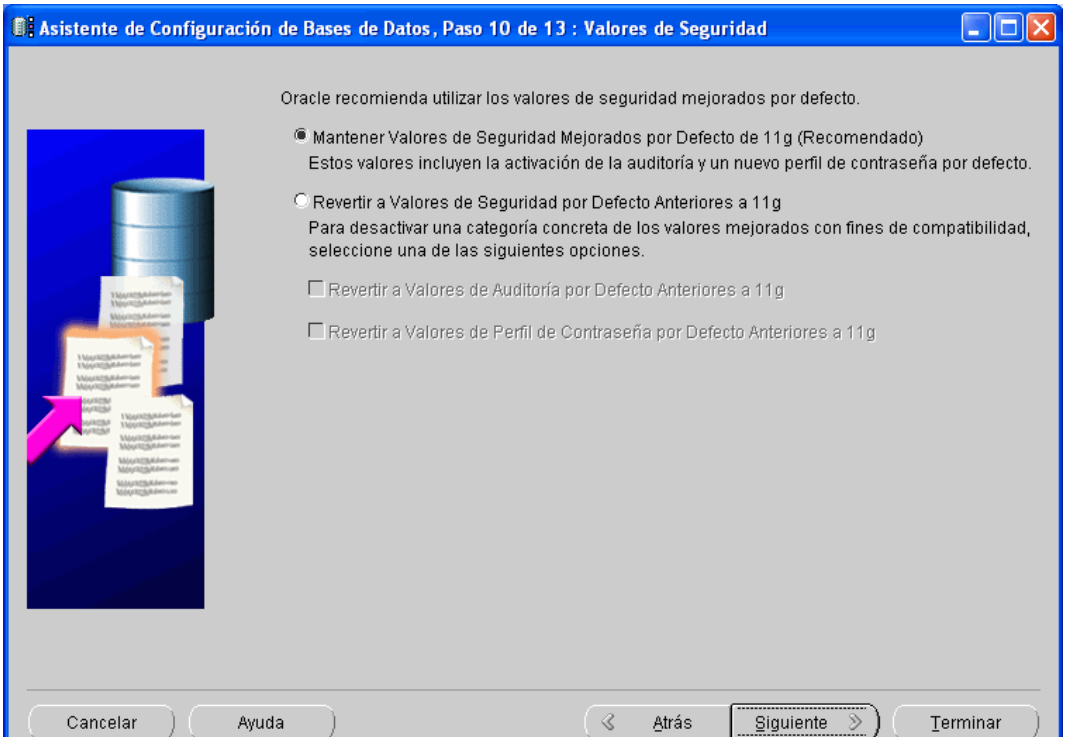

Imagen 31. Proceso de instalación de Oracle 11g: Valores de seguridad

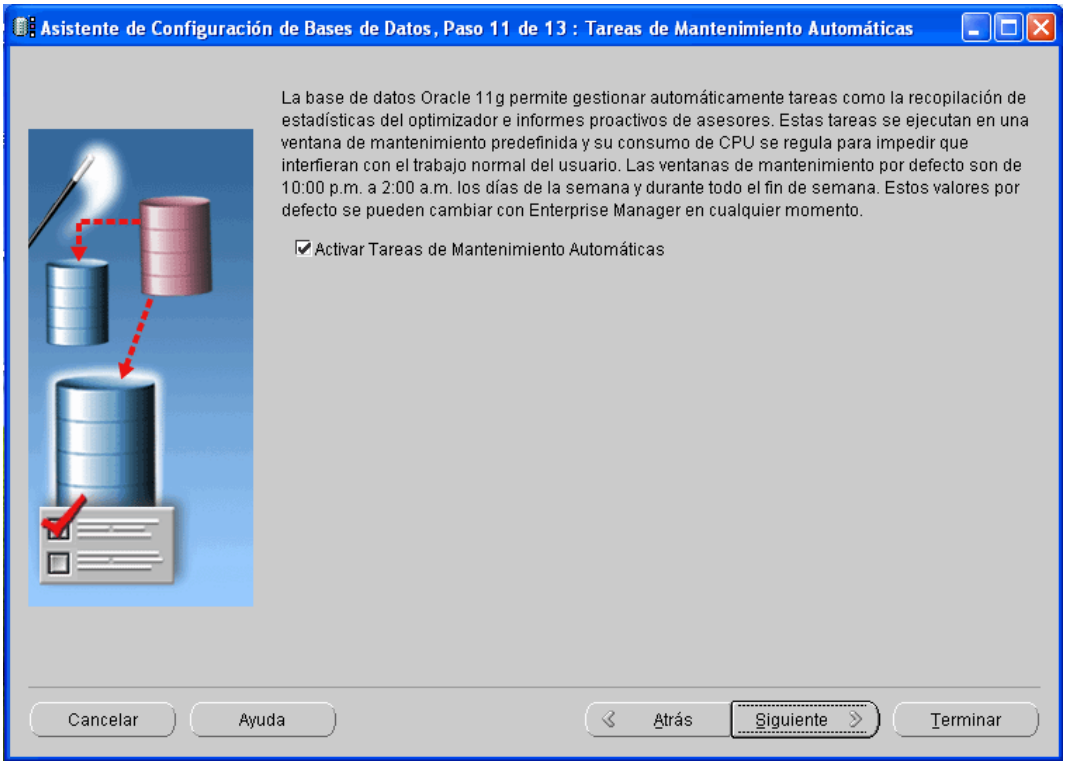

Imagen 32. Proceso de instalación de Oracle 11g: Tareas de mantenimiento automáticas

En el paso 11, al pulsar el botón “Siguiente” aparecerá la siguiente pantalla en la que se podrán crear y suprimir parámetros de almacenamiento de la base de datos, como pueden ser *Tablespaces*, *Archivos de Datos*, *Segmentos de Rollback*…

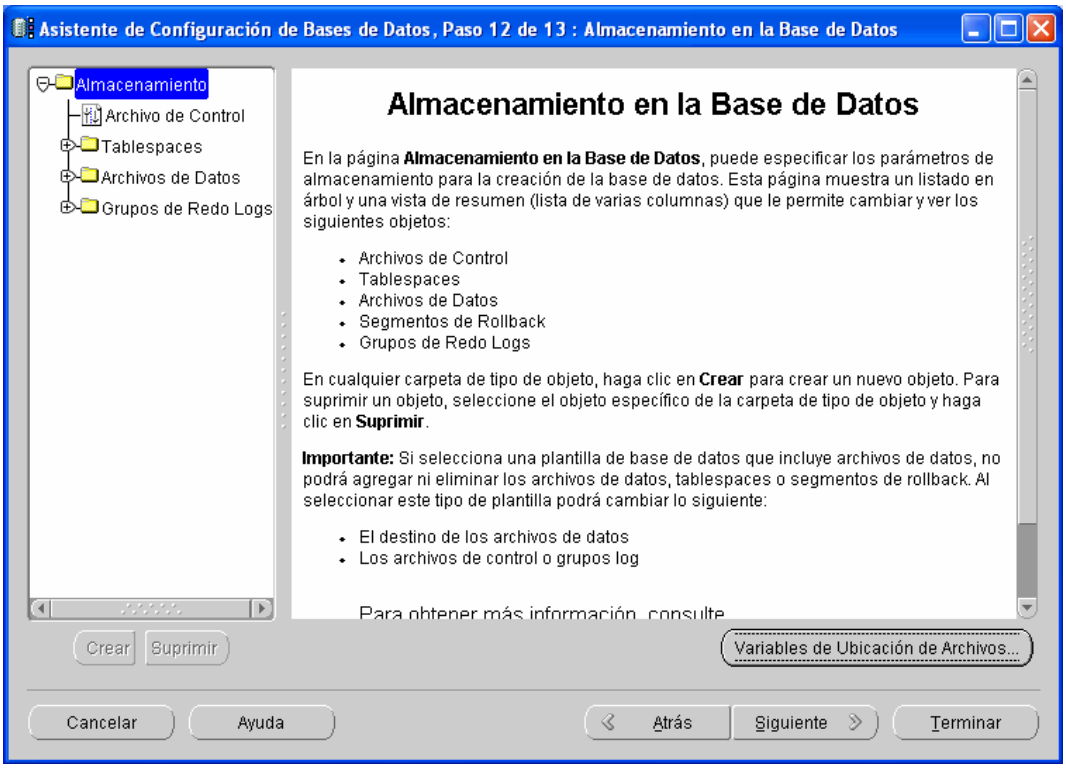

Imagen 33. Proceso de instalación de Oracle 11g: Almacenamiento en la Base de Datos

Aunque en esta pantalla no se cree o suprima algún parámetro de almacenamiento. Posteriormente, una vez finalizada la instalación, también se podrán modificar los parámetros de almacenamiento de la base de datos. En nuestro caso se dejarán los parámetros creados por defecto de Oracle y se pulsará el botón “Terminar”.

Finalmente, en el último paso, se pueden seleccionar las siguientes opciones a la hora de crear la base de datos:

- Crear Base de Datos
- Guardar como Plantilla de Base de Datos
- Generar Archivos de Comandos de Creación de Base de Datos

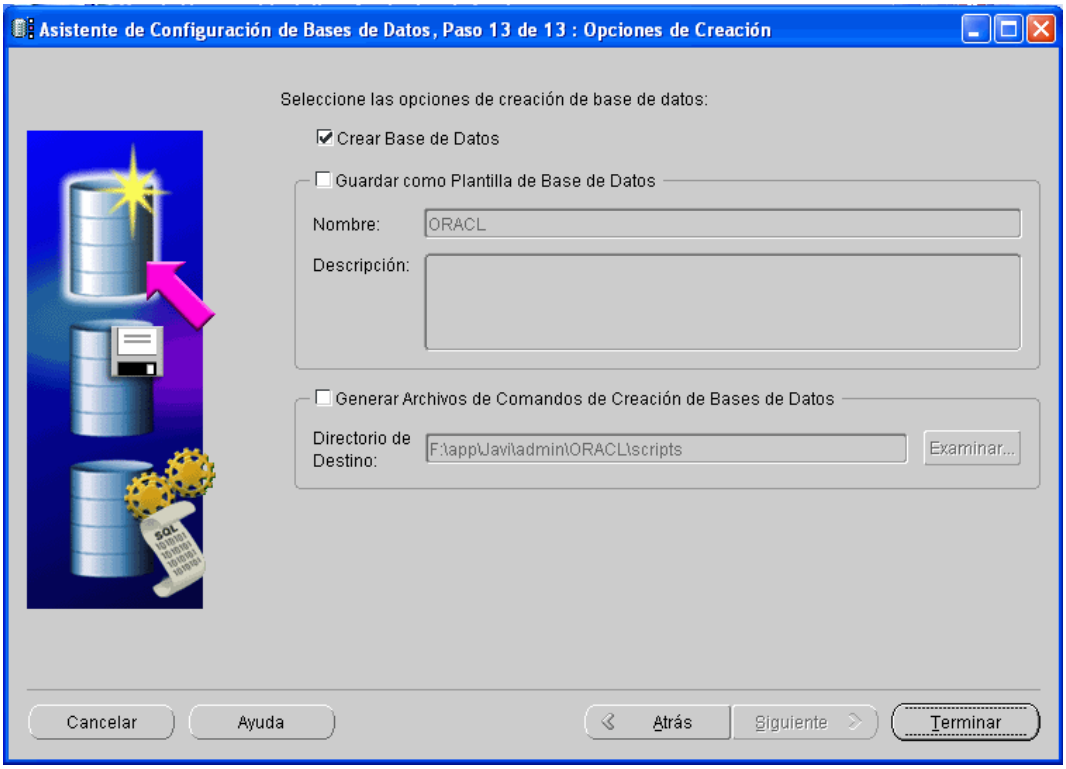

Imagen 34. Proceso de instalación de Oracle 11g: Opciones de creación

Una vez indicada la opción que se desea llevar a cabo, se pulsará el botón “Terminar”.

Antes de que el asistente comience a generar la base de datos, se mostrará una pantalla resumen en la que se especifican las opciones de base de datos que se van a instalar.

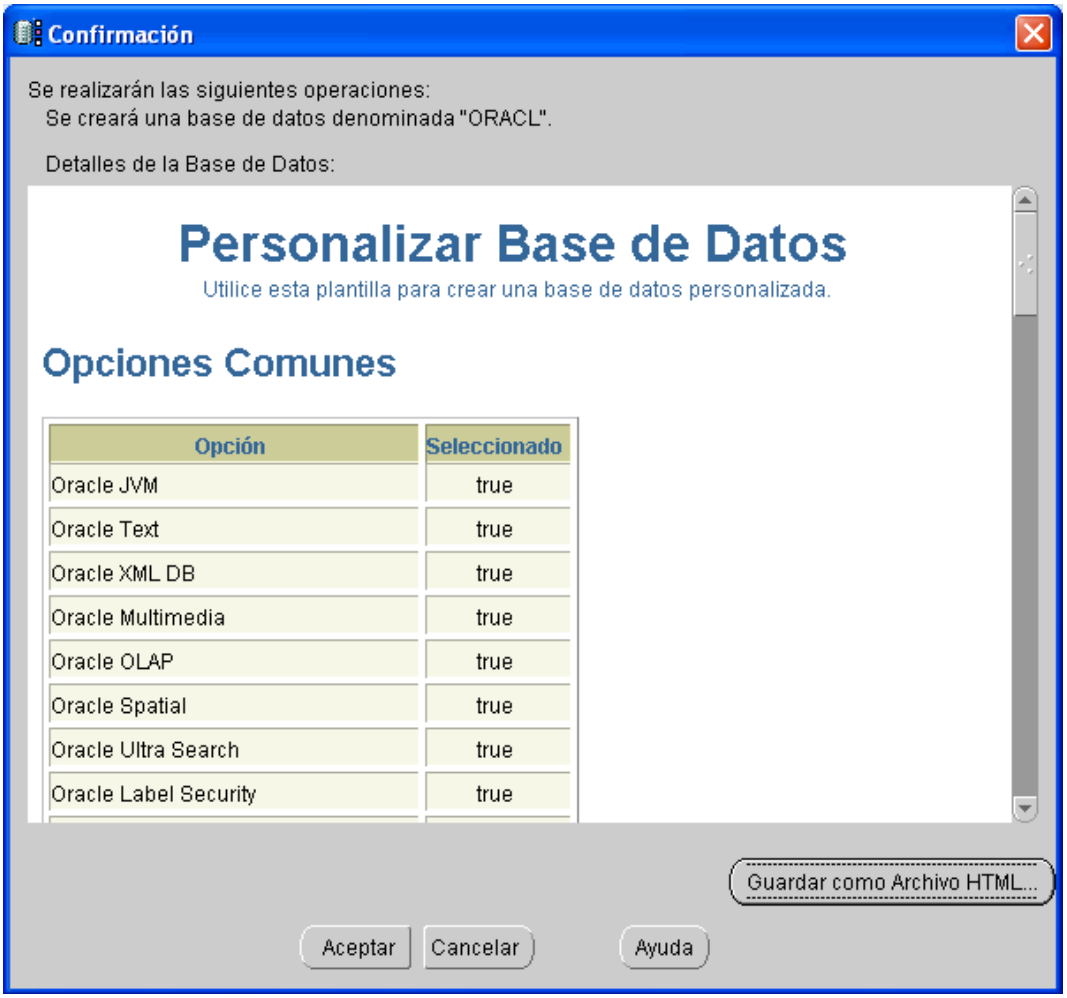

Imagen 35. Proceso de instalación de Oracle 11g: Resumen de opciones de la Base de Datos

Antes de pulsar el botón “Aceptar”, se aconseja revisar que todas las opciones de instalación que están seleccionadas son las correctas, o si falta alguna opción por marcar. En caso de que la configuración no sea correcta, deberá pulsarse el botón “Cancelar” y comenzar de nuevo con la configuración de la base de datos.

Para finalizar, el asistente nos mostrará una pantalla en la que se indica el porcentaje de creación de base de datos que se ha llevado a cabo hasta el momento.

Imagen 36. Proceso de instalación de Oracle 11g: Evolución del proceso de creación de la Base de Datos

Una vez haya finalizado este proceso, la base de datos estará operativa para empezar a ser utilizada.

Capítulo 6: Nociones básicas de Oracle Label Security

Introducción

Oracle Label Security brinda capacidades amplias y flexibles para el control de acceso basado en etiquetas, a fin de respaldar la implementación de aplicaciones de seguridad de múltiples niveles.

A continuación, se exponen las nociones básicas de Oracle Label Security.

Etiquetas de sensibilidad y mediación de acceso

La mediación de acceso de Oracle Label Security funciona al comparar una etiqueta de sensibilidad con las autorizaciones de etiquetas asignadas a un usuario de aplicaciones.

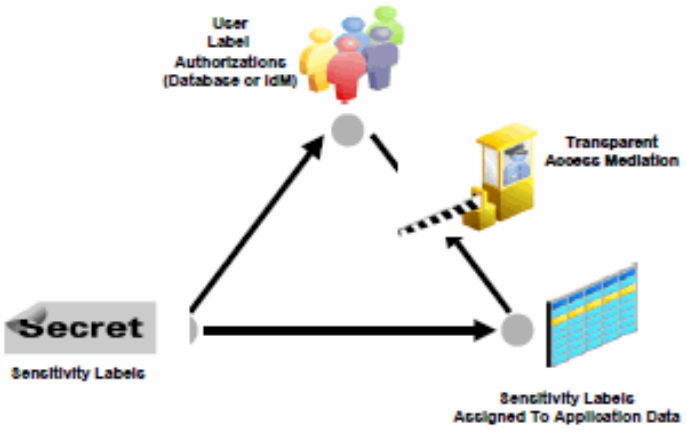

Imagen 37. Gráfico de mediación de acceso de Oracle Label Security

Fuente: Oracle Label Security – Mejores Prácticas para Aplicaciones de Gobierno y Defensa

Las etiquetas de sensibilidad ofrecen controles sofisticados y determinan el acceso de un usuario de aplicaciones a los datos de aplicaciones. En el capítulo *"Calidad y seguridad en Oracle"*, se puede encontrar toda la información referente a la composición y representación externa de las etiquetas de sensibilidad.

Algoritmo de lectura para mediación de acceso

Oracle Label Security media el acceso al comparar las autorizaciones de las etiquetas de usuarios con un nivel de sensibilidad.

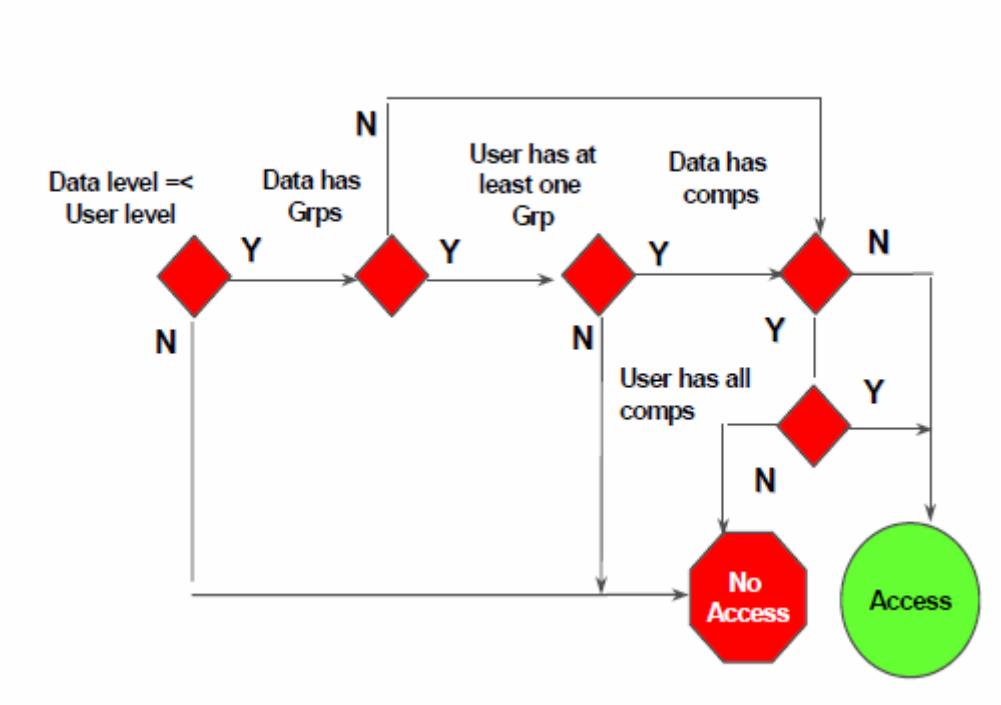

Imagen 38. Algoritmo de lectura para mediación de acceso

Fuente: Oracle Label Security – Mejores Prácticas para Aplicaciones de Gobierno y Defensa

Incorporación de Oracle Label Security a una aplicación

A continuación se exponen los diferentes pasos a realizar para implementar con éxito una política Oracle Label Security.

1 – Analizar el esquema de la aplicación

Analizar la aplicación implica identificar las tablas que necesitan aplicar una política Oracle Label Security. En la mayoría de los casos, un pequeño porcentaje de las tablas de aplicaciones necesitará una política Oracle Label Security.

2 – Analizar los niveles de sensibilidad

Una vez se hayan identificado las tablas en las que se va aplicar Oracle Label Security, los datos contenidos en las mismas deben ser evaluados. Sería conveniente la ayuda de alguien con un profundo entendimiento de los datos. Los niveles de datos se refieren al nivel de sensibilidad de los datos. Ejemplo de niveles de sensibilidad: *"No Clasificado"*, *"Sensible"* y *"Altamente Sensible"*.

3 – Analizar los grupos de datos

La definición de grupos es útil para controlar el acceso basado en la propiedad de datos y para compartir los datos en todas las organizaciones. Al igual que en la definición de Niveles de Sensibilidad, es aconsejable disponer de la ayuda de alguien con amplia familiaridad con las operaciones de negocio. Ejemplo de grupos: *"Estados Unidos"*, *"Europa"*, *"Asia"*…

4 – Analizar los compartimentos de datos

La definición de compartimentos de datos se utiliza principalmente en el ámbito de gobierno y defensa. Una aplicación comercial puede considerar que los grupos de datos son suficientes para brindar el nivel necesario de control de acceso. No obstante, las organizaciones comerciales pueden considerar que los compartimentos son útiles para etiquetar información personalmente identificable. En los ámbitos de gobierno y defensa, los compartimentos son típicamente utilizados para restringir el acceso a proyectos o áreas especializadas de conocimiento.

5 – Analizar la población de usuarios

Este paso requiere obtener un entendimiento de la gran cantidad de roles y responsabilidades implicados en la población de usuarios. Después de que la población de usuarios se ha separado en uno o más roles o áreas funcionales, se debe realizar una comparación entre los niveles de datos que han sido identificados en el paso 2 y las autorizaciones de etiquetas de la población de usuarios.

6 – Analizar las autorizaciones especiales

Oracle Label Security dispone de varias autorizaciones especiales que pueden ser asignadas a los usuarios y procedimientos almacenados. En este paso, debe examinarse a los usuarios administrativos y altamente privilegiados y determinarse si alguna de las autorizaciones especiales de Oracle Label Security debería ser asignada al usuario.

7 – Revisar y documentar

El paso final antes de comenzar la implementación es revisar y documentar la información recopilada. El documento resultante debería ser incluido como parte de la política de seguridad de la empresa.

Implementación

La implementación puede realizarse utilizando Oracle Enterprise Manager o la interfaz de línea de comando de Oracle Label Security. Para exponer de una forma más práctica la implementación de Oracle Label Security, inicialmente se expondrán los puntos a seguir y a continuación, se mostrará la implementación de un caso práctico.

1. Inicialmente deberán realizarse los pasos de análisis anteriormente mencionados.

2. Crear la política Oracle Label Security.

3. Definir todos los componentes para etiquetas de datos válidos utilizando niveles, compartimentos y grupos con la información recopilada durante el análisis.

4. Crear etiquetas de datos válidos para la política mediante el uso de los componentes definidos en el punto anterior.

5. Asignación de autorizaciones de etiquetas a las poblaciones de usuarios y autorizaciones especiales adecuadas de Oracle Label Security utilizando los datos recopilados durante el análisis.

6. Aplicar la política a la tabla de aplicaciones.

   En caso de haber datos antiguos en la tabla, puede ser necesario aplicar la política a la tabla con la opción *"No aplicar el control inicialmente"*. Una vez que la política sea aplicada a la tabla, la etiqueta de sensibilidad estará vacía y ningún dato estará visible.

7. Actualizar datos antiguos con las etiquetas adecuadas de datos.

8. Si se aplicó la política pero está deshabilitada, debe activarse la política Oracle Label Security.

Caso práctico

Se va a disponer de una pequeña aplicación encargada de gestionar las nóminas de una empresa. Toda nómina registrada en el sistema podrá ser consultada por los usuarios, pero solo los miembros de Recursos Humanos serán los que puedan ver la cuantía a pagar. Para solventar este problema de una forma rápida y eficaz, se ha pensado crear una política "Oracle Label Security".

Al tratarse de un pequeño ejemplo, la aplicación solamente va a tener los tres usuarios detallados a continuación:

HR_APP: Administrador de la aplicación y encargado de gestionar la política "Oracle Label Security". (Usuario de BBDD)

PRUEBA: Personal perteneciente a recursos humanos.

PRUEBA2: Personal no perteneciente a recursos humanos.

1. En este ejemplo el proceso de análisis es muy sencillo. Se deberá crear una política Oracle Label Security definiendo dos niveles de sensibilidad de los datos.

Posteriormente, se deberán crear los usuarios necesarios y asociarles las autorizaciones correspondientes.

Finalmente, se deberá crear una política de base de datos virtual, que será la que informe si un usuario tiene permisos para ver la cuantía de las nóminas o no.

2. Crear la política Oracle Label Security.

En el ejemplo que se muestra a continuación, se ha utilizado Oracle Enterprise Manager para la creación de la Política de Seguridad.

I. Acceder a Oracle Enterprise Manager con el usuario *"LBACSYS"*.

II. En la pestaña “Servidor” pulsar el enlace “Oracle Label Security”.

Imagen 39. Oracle Enterprise Manager. Pestaña Servidor

III. Pulsar el botón “Crear”.

Imagen 40. Oracle Enterprise Manager. Políticas de Label Security

IV. Definir nombre de la política y la columna de etiqueta.

Imagen 41. Oracle Enterprise Manager. Crear Política de Label Security (Pestaña General)

Al pulsar el botón "Aceptar" la Política Oracle Label Security se creará.

3. Definir todos los componentes para etiquetas de datos válidos utilizando niveles, compartimentos y grupos con la información recopilada durante el análisis.

I. Una vez creada la política, debe pulsarse sobre la pestaña "Componentes de las Etiquetas" para proceder a crear los mismos.

Imagen 42. Oracle Enterprise Manager. Crear Política de Label Security (Componentes de las Etiquetas)

II. En el apartado niveles, se crearán los niveles *"Confidential"* y *"Sensitive"*.

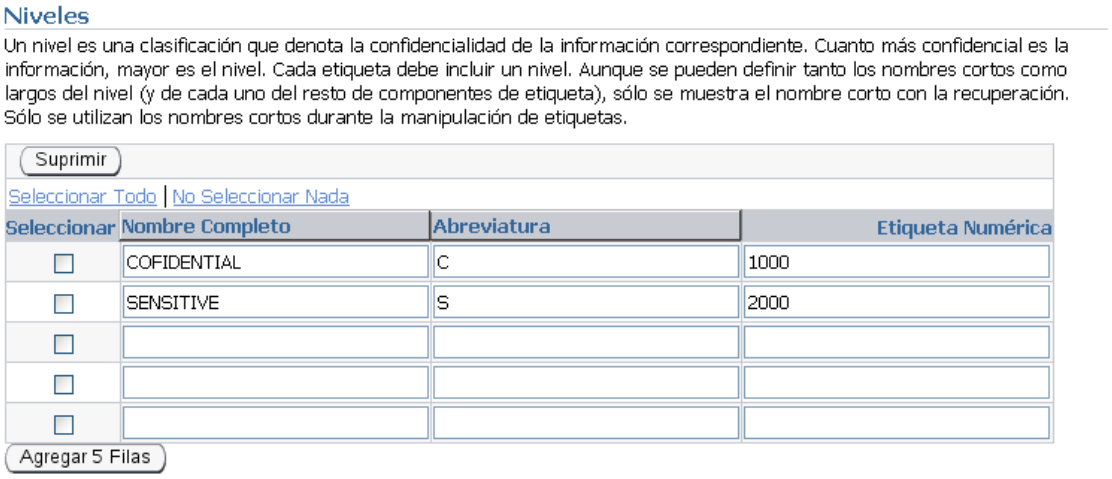

Imagen 43. Oracle Enterprise Manager. Creación de niveles de la política

III. En el apartado compartimentos, se creará el compartimento *"PERS_INFO"*.

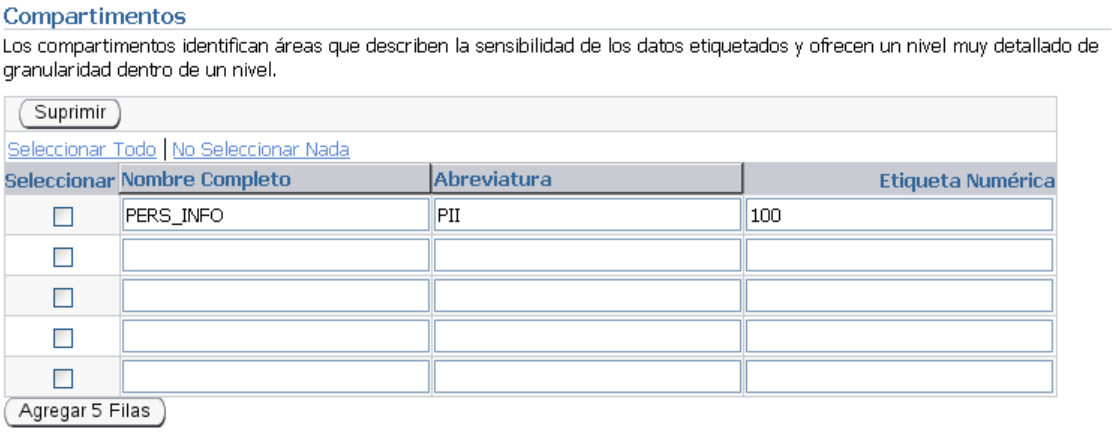

Imagen 44. Oracle Enterprise Manager. Creación de compartimentos

Una vez se hayan elaborado los niveles, compartimentos y grupos necesarios, pulsando el botón "Aceptar" estos se crearán en la política Oracle Label Security.

4. Para nuestro ejemplo no es necesario crear etiquetas de datos válidos.

5. Asignación de autorizaciones de etiquetas a las poblaciones de usuarios y autorizaciones especiales adecuadas de Oracle Label Security utilizando los datos recopilados durante el análisis.

Antes de comenzar con la asociación de autorizaciones a usuarios, se deberá crear el usuario *"HR_APP"* a través de SQL*PLUS. Al ejecutar SQL*PLUS habrá que identificarse como *"SYSTEM"* y lanzar el siguiente script:

```
Drop user hr_app cascade;        /* Eliminación del usuario hr_app */
Create user hr_app               /* Creación del usuario hr_app */
Identified by administrador
Default tablespace ejemplos
Temporary tablespace temp
Quota unlimited on ejemplos;
Grant create session to hr_app; /* Asignación del permiso creación de sesión */
Create table hr_app.nominas as /* Creación de la table hr_app.nominas */
Select * from nominas;       /* como una copia de la tabla nominas  */
```

I. Para asociar un usuario, al editar la política deberá pulsarse el botón "Agregar Usuarios".

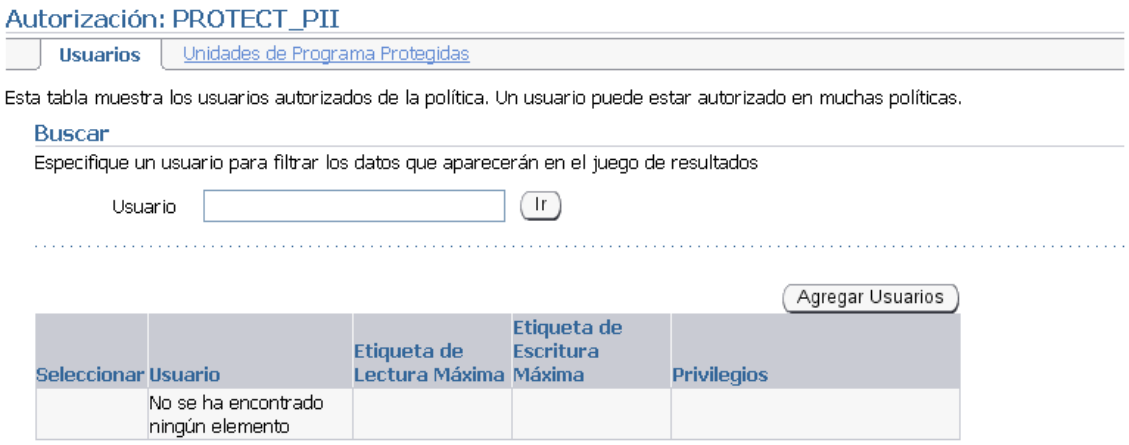

Imagen 45. Oracle Enterprise Manager. Agregar usuarios a la política

II. Al pulsar el botón "Agregar Usuarios" se mostrará la siguiente pantalla.

Imagen 46. Oracle Enterprise Manager. Agregar usuarios

A continuación, para agregar un usuario, habrá que pulsar el botón "Agregar".

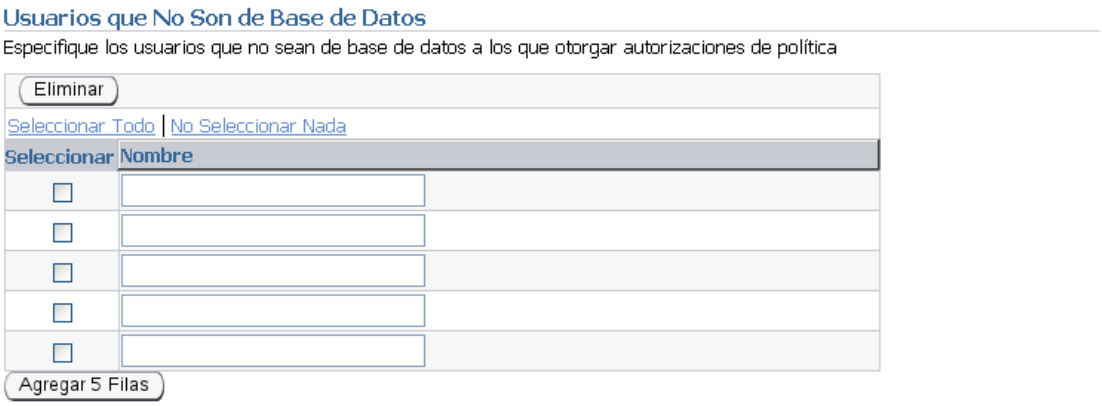

Imagen 47. Oracle Enterprise Manager. Listado de usuarios a agregar

En este listado, se especificará el usuario o usuarios deseados. En nuestro caso, se introducirá cada usuario a la vez que se le asocia el nivel al que va a pertenecer.

Definición del usuario *"PRUEBA"*:

Una vez introducido el usuario *"Prueba"*, se deberá seleccionar y pulsar el botón "Siguiente" hasta llegar a la pestaña "Niveles, Compartimentos y Grupos".

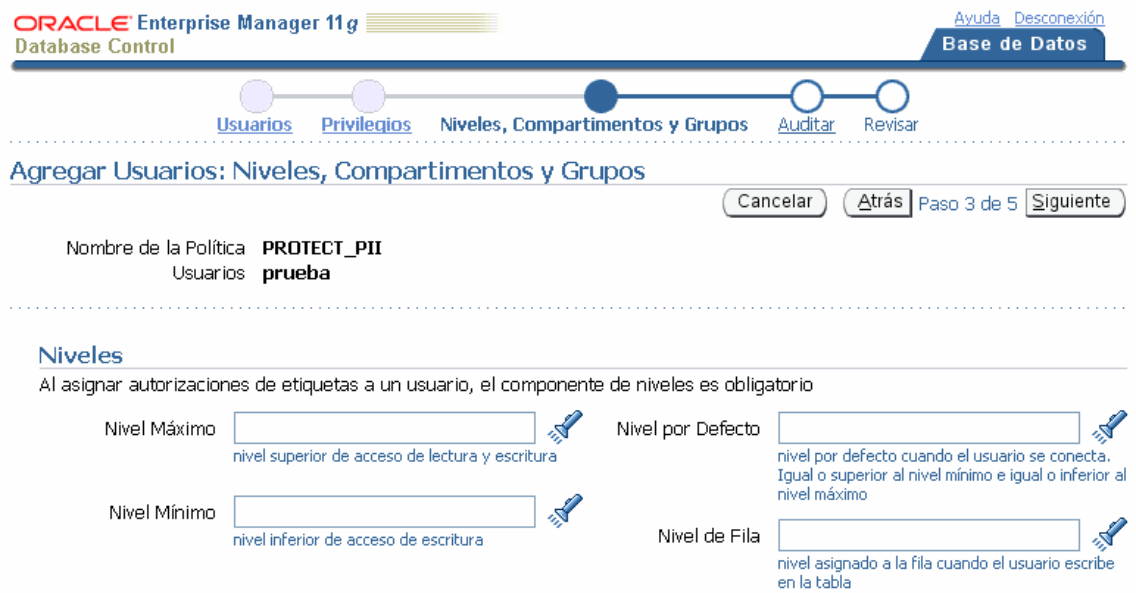

Imagen 48. Oracle Enterprise Manager. Definición de la autorización del usuario *"Prueba"*

En está pantalla se especificarán los niveles del usuario, siendo: *"Nivel Máximo: Sensitive"*, *"Nivel por Defecto: Sensitive"*, *"Nivel Mínimo: Confidential"* y *"Nivel de Fila: Confidential"*.

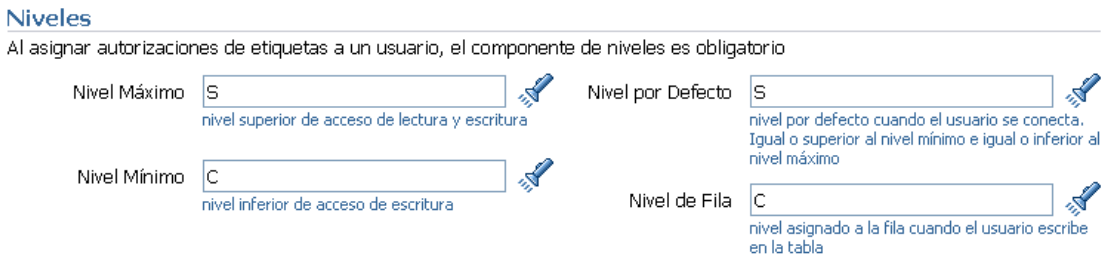

Imagen 49. Oracle Enterprise Manager. Autorización del usuario *"Prueba"*

A continuación, se asociará al usuario el compartimento al que pertenece, en caso de pertenecer a alguno.

Para este usuario, se buscará el compartimento *"PERS_INFO: PII"* definido en el apartado anterior.

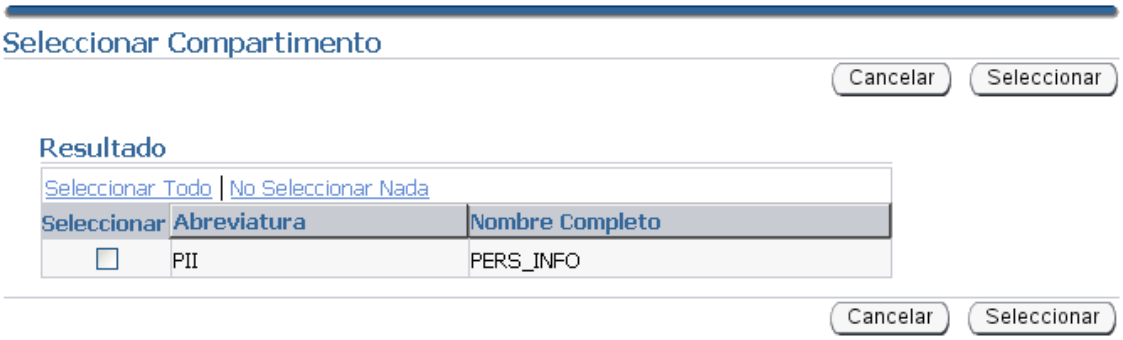

Imagen 50. Oracle Enterprise Manager. Compartimento a asociar al usuario

Una vez encontrado, se pulsará el botón "Seleccionar". Dicho compartimento aparecerá en la sección de compartimentos del usuario.

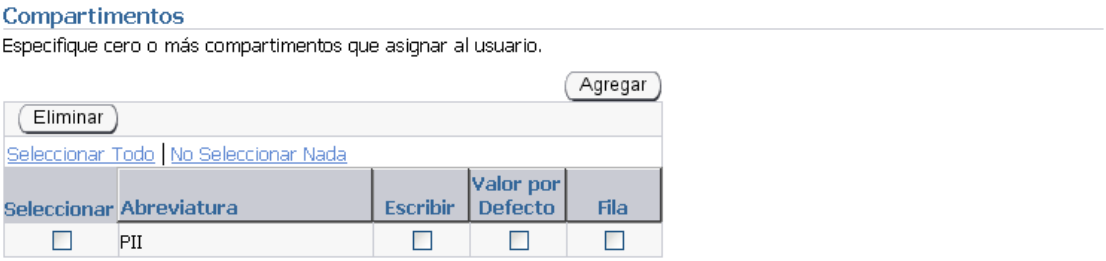

Imagen 51. Oracle Enterprise Manager. Compartimento asociado al usuario

Seguidamente, se marcará “Valor por Defecto” del compartimento para este usuario.

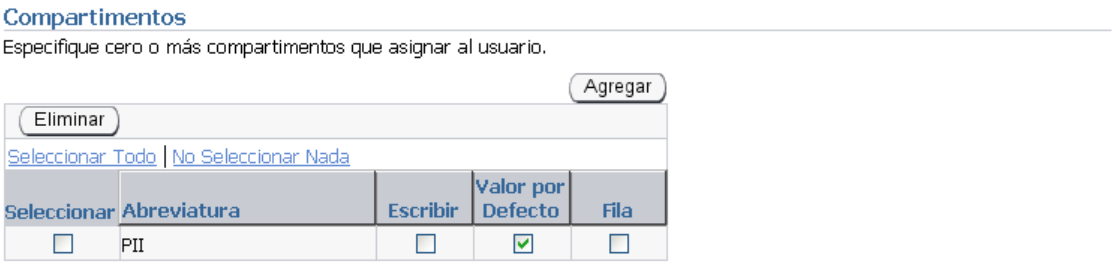

Imagen 52. Oracle Enterprise Manager. Compartimento asociado al usuario: Valor por defecto

Finalmente, se pulsará el botón "Siguiente" y aparecerá la siguiente pantalla resumen del usuario a crear.

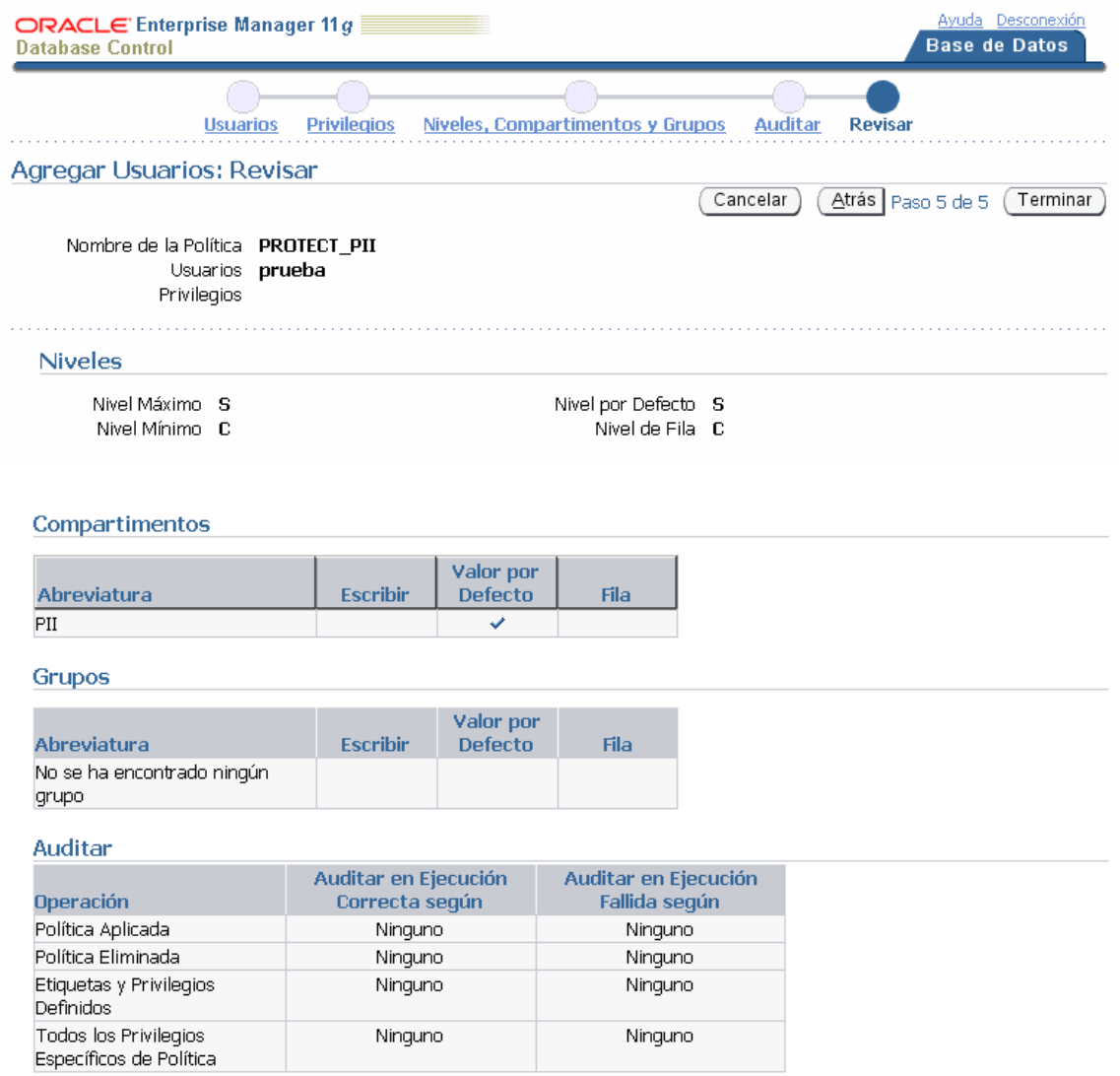

Imagen 53. Oracle Enterprise Manager. Resumen propiedades del usuario *"Prueba"*

Una vez se haya comprobado que todas las especificaciones son correctas se pulsará el botón "Terminar" para crear el usuario y asociarle los niveles y compartimento descrito anteriormente.

Si el usuario se ha creado correctamente, Oracle Enterprise Manager mostrará la siguiente pantalla.

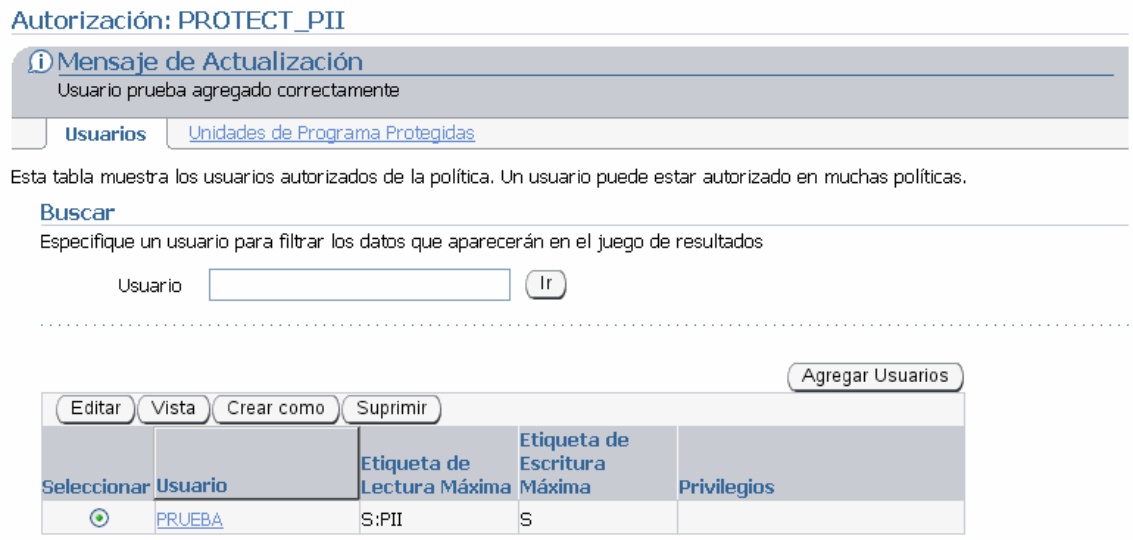

Imagen 54. Oracle Enterprise Manager. Usuario *"Prueba"* creado correctamente

Definición de usuario *"PRUEBA2"*:

Una vez introducido el usuario *"Prueba2"*, se seleccionará y se pulsará el botón "Siguiente" hasta llegar a la pestaña "Niveles, Compartimentos y Grupos".

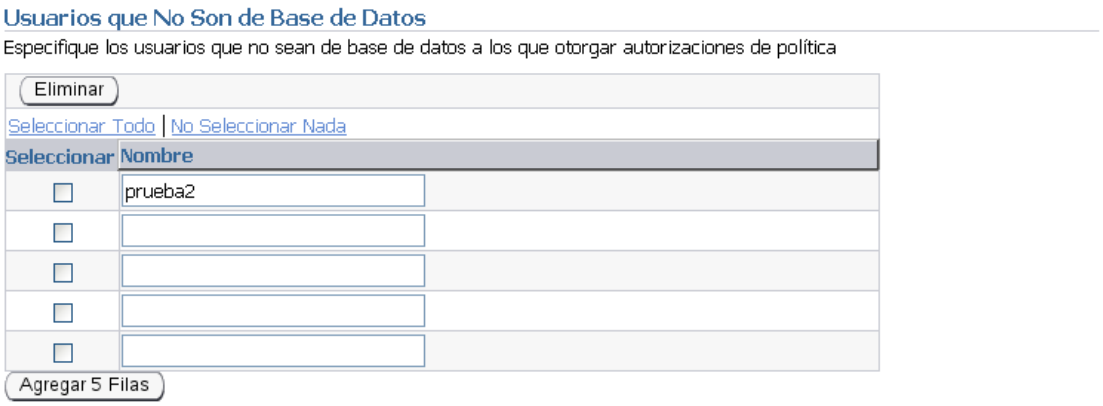

Imagen 55. Oracle Enterprise Manager. Listado de usuarios a agregar (Usuario: *"Prueba2"*)

En este caso se especificará lo siguiente en los niveles del usuario: *"Nivel Máximo: Confidential"*, *"Nivel por Defecto: Confidential"*, *"Nivel Mínimo: Confidential"* y *"Nivel de Fila: Confidential"*.

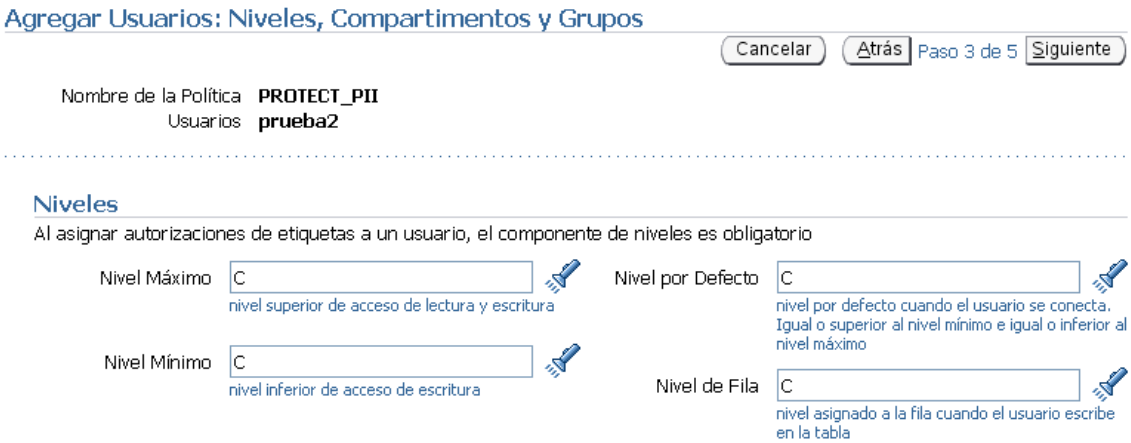

Imagen 56. Oracle Enterprise Manager. Autorización del usuario *"Prueba2"*

Este usuario no pertenece a ningún compartimento, por tanto, se pulsará el botón "Siguiente" hasta que aparezca la pantalla resumen del usuario a crear.

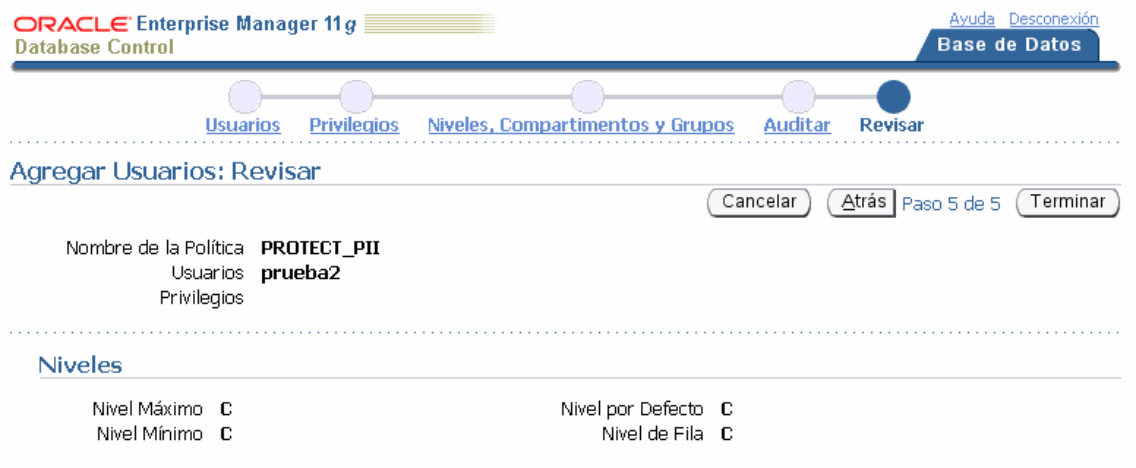

Imagen 57. Oracle Enterprise Manager. Resumen propiedades del usuario *"Prueba2"*

Se pulsará el botón "Terminar" para crear el usuario y asociarle los niveles y compartimento descrito anteriormente.

Si el usuario se ha creado correctamente, Oracle Enterprise Manager mostrará la siguiente pantalla.

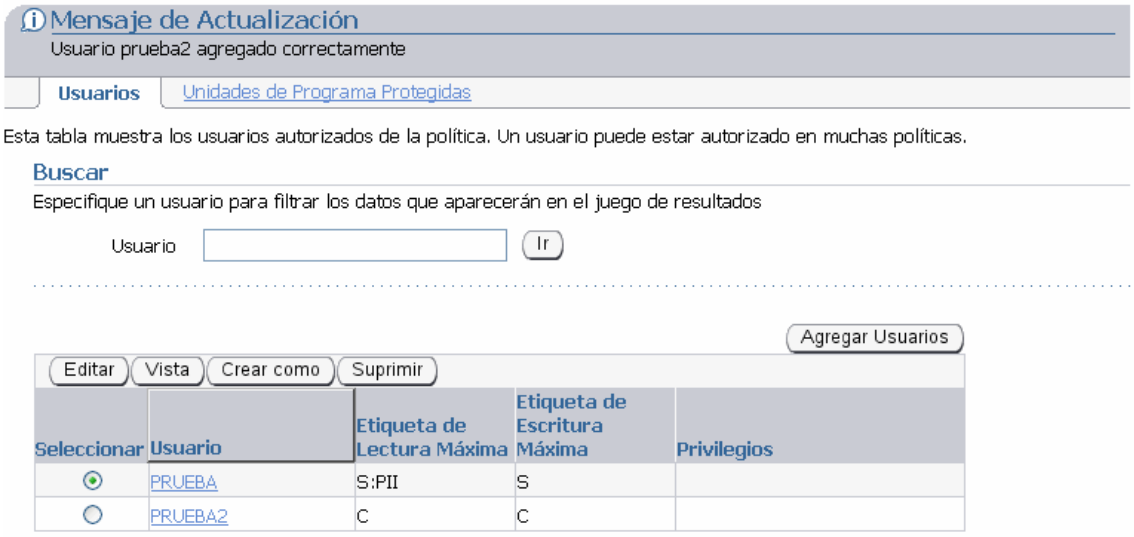

Imagen 58. Oracle Enterprise Manager. Usuario *"Prueba2"* creado correctamente

Asociar al usuario *"HR_APP"* los permisos de administración de la política Oracle Label Security.

El usuario deberá editar la política y en el apartado "Usuarios de Bases de Datos" pulsar el botón "Agregar".

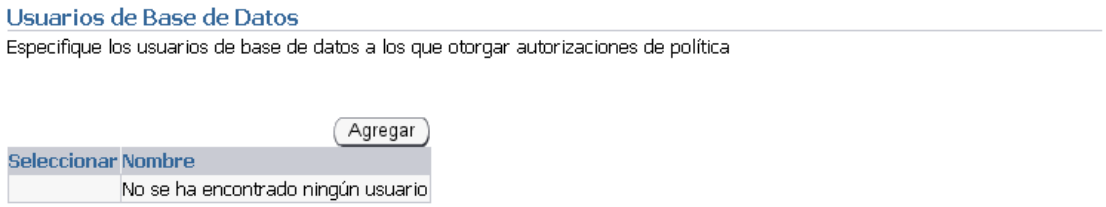

Imagen 59. Oracle Enterprise Manager. Agregar usuario de base de datos

Aparecerá la siguiente pantalla de búsqueda.

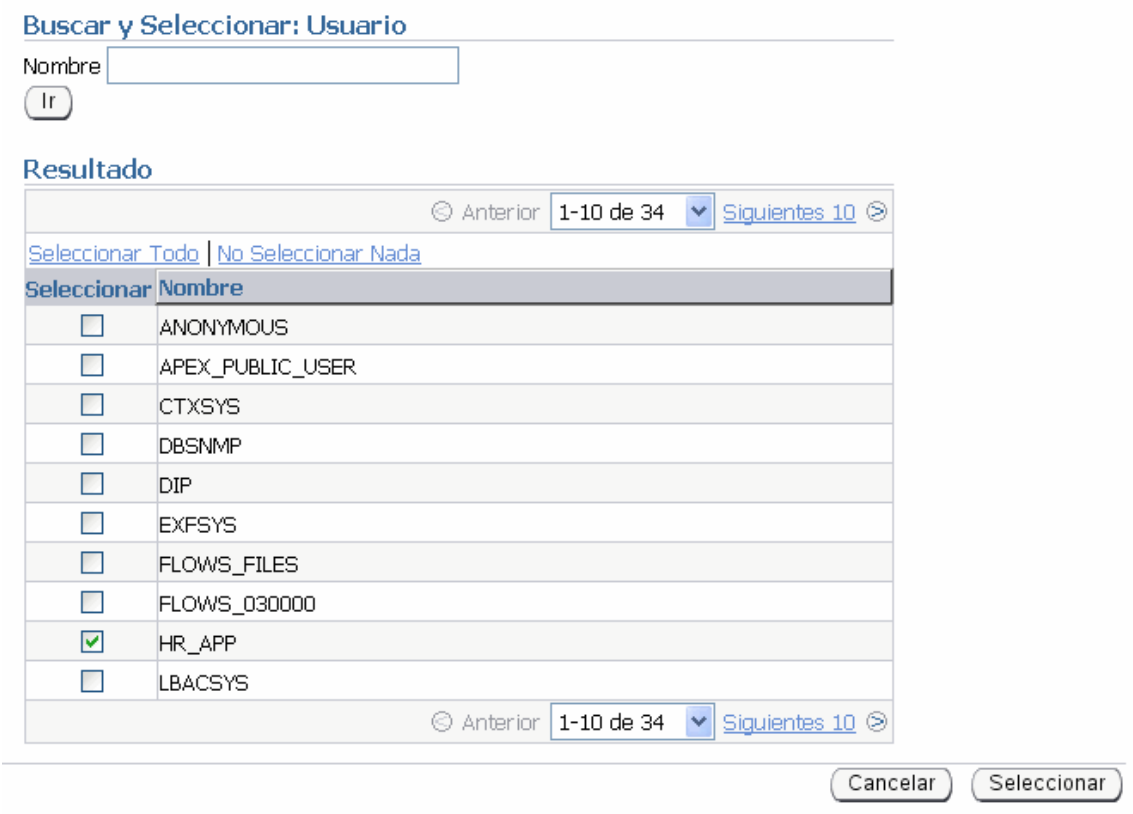

Imagen 60. Oracle Enterprise Manager. Búsqueda de usuario

Se marcará el usuario *"HR_APP"* y se pulsará el botón "Seleccionar".

Una vez seleccionado el usuario, este aparecerá en el apartado de "Usuarios de Bases de Datos".

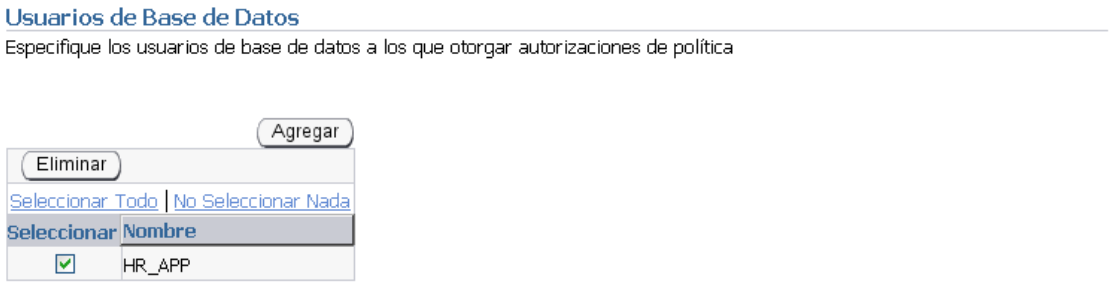

Imagen 61. Oracle Enterprise Manager. Selección del usuario de base de datos especificado

Se seleccionará el usuario y se pulsará el botón "Siguiente" hasta llegar a la pestaña "Privilegios".

En esta pestaña se asociara el privilegio *"Asumir perfil de otro usuario mediante set_access_profile (PROFILE_ACCESS)"*.

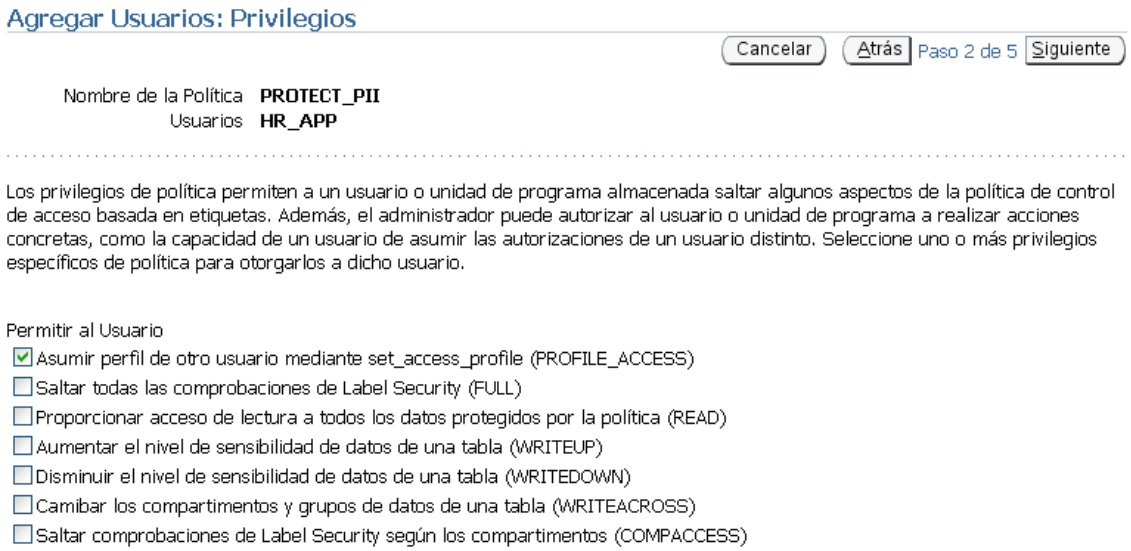

Imagen 62. Oracle Enterprise Manager. Asignación de privilegios al usuario *"HR_APP"*

Se pulsará el botón “Siguiente” hasta llegar a la pantalla resumen.

Imagen 63. Oracle Enterprise Manager. Resumen propiedades del usuario *“HR_APP”*

Finalmente, se pulsará el botón “Terminar”.

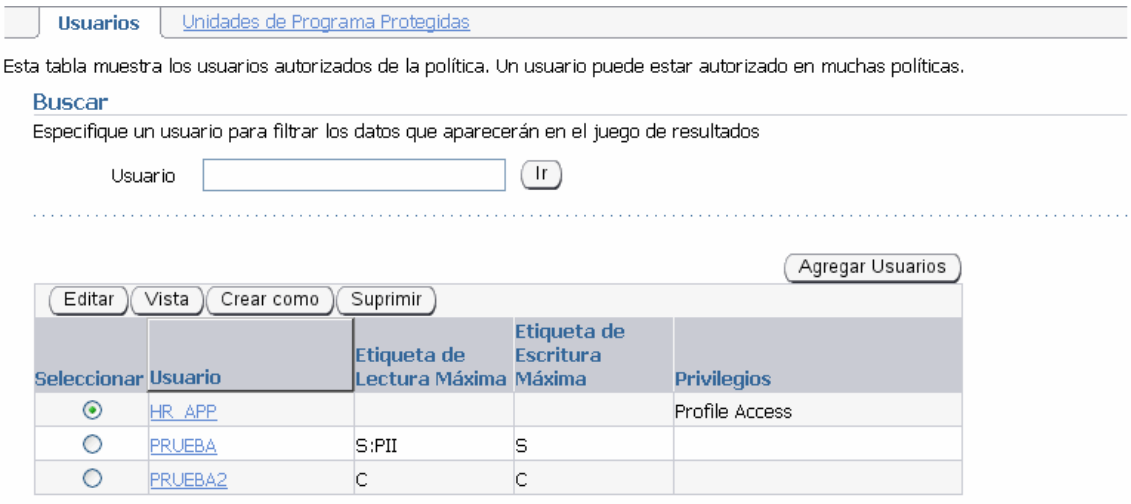

Imagen 64. Oracle Enterprise Manager. Resumen de usuarios de la política

6. Aplicar la política a la tabla de aplicaciones.

Anteriormente, se ha creado la asociación de etiquetas a usuarios. Ahora, se debe crear la política de la Base de Datos Virtual Privada (VPD). La política VPD hará lo siguiente:

Obtener el código numérico de la etiqueta asociada al usuario actual.

Obtener el código numérico de la etiqueta *"S:PII"*.

Si la etiqueta de usuario ≥ *"S:PII"* permite el acceso a todas las filas con columnas sensibles.

Si la etiqueta de usuario < *"S:PII"* permite el acceso a todas las filas, pero el valor de las columnas sensibles *"PII"* será nulo.

I. Creación de la función asociada a la política VPD.

Para llevar a cabo la creación de la política VPD, habrá que acceder a SQL*PLUS con el usuario *LBACSYS* y ejecutar el siguiente script:

```
CREATE OR REPLACE FUNCTION f_protect_pii (schema in varchar2, tab in
varchar2)
 RETURN varchar2 AS
  predicate          varchar2(2000) default '1=2';
  session_lab        varchar2(4000) default null;
  session_tag        number;
  sens_tag           number;
 BEGIN
 session_lab := sa_session.label('PROTECT_PII');
 if session_lab is not null then
   session_tag := char_to_label('PROTECT_PII',session_lab);
 else null;
 end if;
 sens_tag := char_to_label('PROTECT_PII','S:PII');
 if dominates (session_tag, sens_tag) = 1 then
   predicate := '1=1';
 else null;
 end if;
 return predicate;
END;
```

II. Creación de la política VPD.

Acceder a Oracle Enterprise Manager con el usuario *LBACSYS*, navegar a la pestaña "Servidor" y pulsar el enlace "Políticas de Bases de Datos Privada".

Imagen 65. Oracle Enterprise Manager. Pestaña Servidor

A continuación, en la ventana de "Políticas de Bases de Datos Privada" pulsar el botón "Crear".

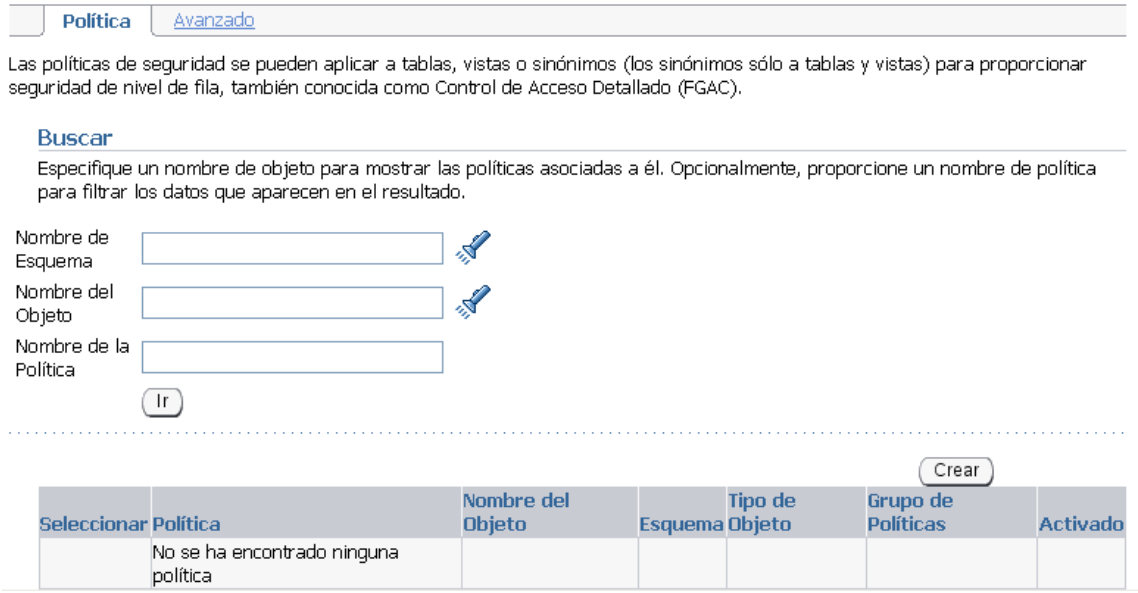

Imagen 66. Oracle Enterprise Manager. Crear política de base de datos privada

Seguidamente, en la sección "General" se deberá introducir el *"Nombre de la Política: vpd_protect_pii"*, *"Nombre del objeto: hr_app.nominas"* y *"Tipo de Política: Context_Sensitive"*.

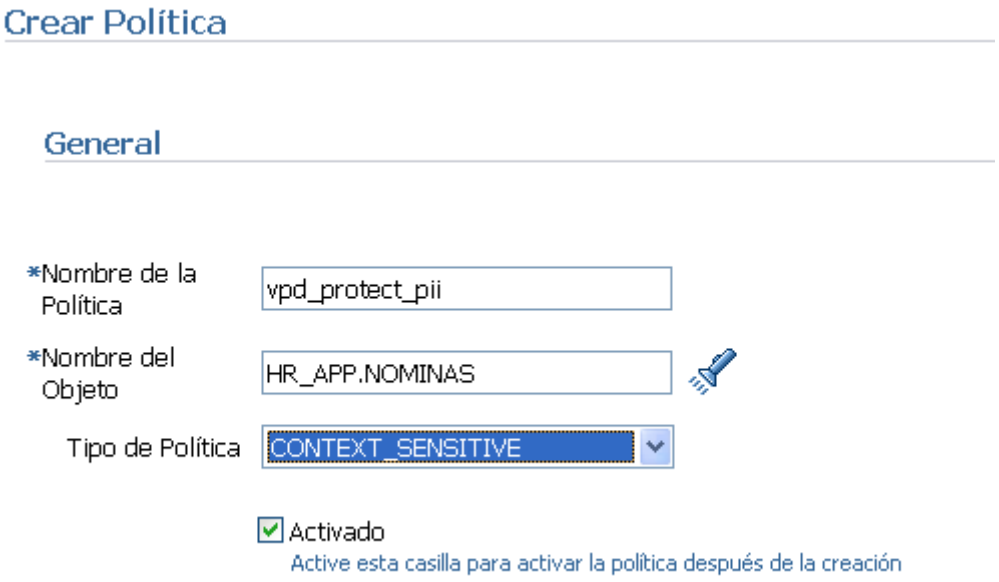

Imagen 67. Oracle Enterprise Manager. Datos de la política de base de datos privada

En la sección "Función de Política", se introducirá en *"Función de Política: Lbacsys.f_protect_pii"*.

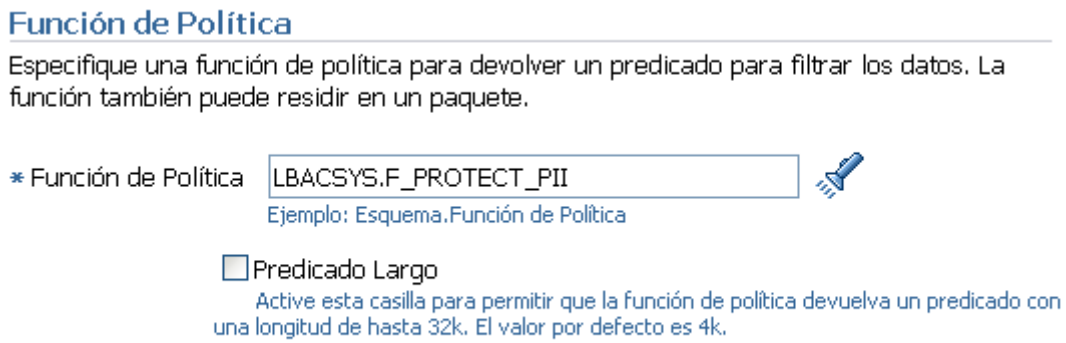

Imagen 68. Oracle Enterprise Manager. Función de política de base de datos privada

En la sección de "Forzado" se marcará la opción *"Select"*.

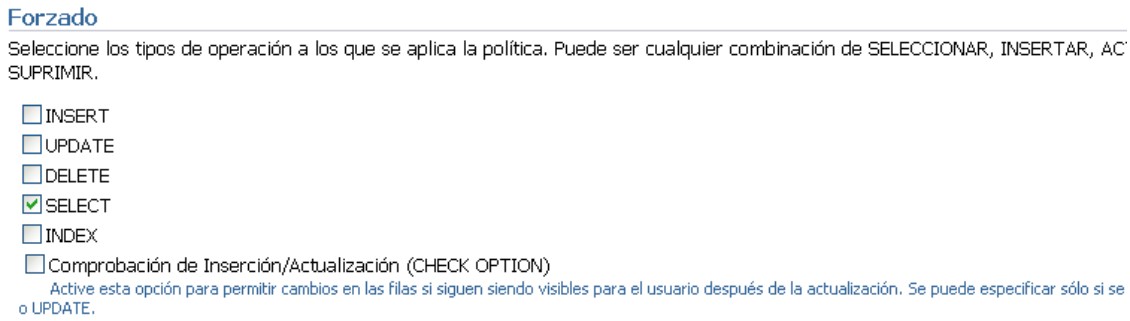

Imagen 69. Oracle Enterprise Manager. Selección de forzado de la política de base de datos privada

En la sección "Columnas Relevantes de Seguridad", se especificará la columna correspondiente a la cuantía de la tabla *"hr_app.nominas"*. Para ello, se pulsará el botón "Agregar".

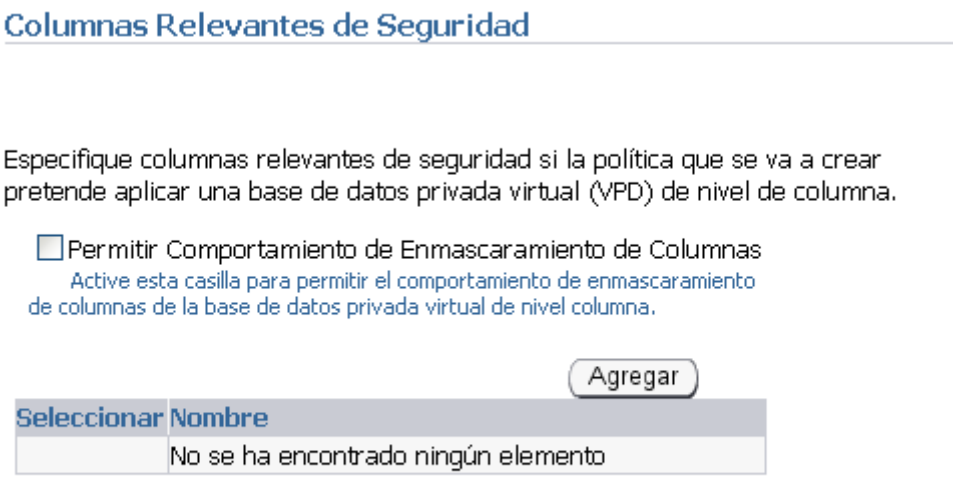

Imagen 70. Oracle Enterprise Manager. Columnas relevantes de seguridad

Se seleccionará la columna *"NM_IMPORTE_FINAL"* correspondiente a la cuantía.

Imagen 71. Oracle Enterprise Manager. Selección columna relevante de seguridad

Y se pulsará el botón "Seleccionar".

Finalmente se marcará la opción *"Permitir Comportamiento de Enmascaramiento de Columnas"* y se pulsará el botón "OK" para finalizar con la creación de la política VPD.

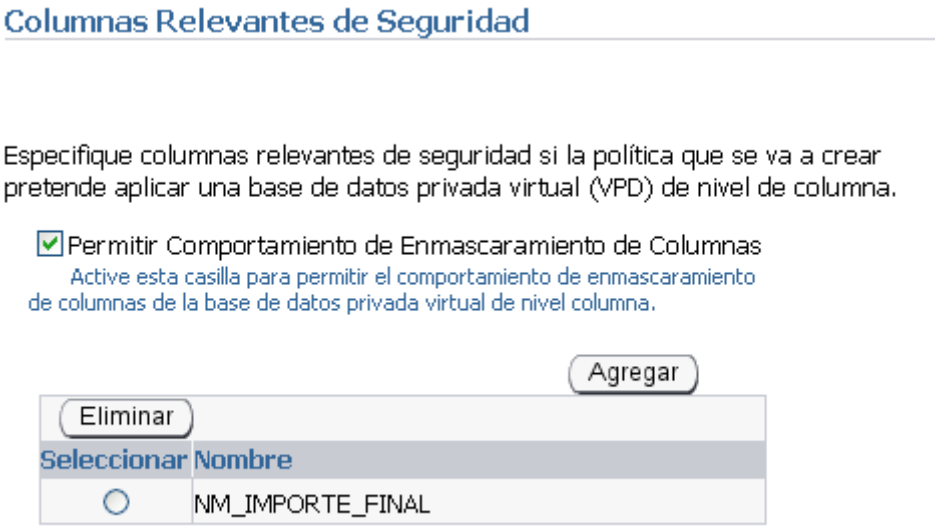

Imagen 72. Oracle Enterprise Manager. Selección de enmascaramiento de columnas

7. Actualizar datos antiguos con las etiquetas adecuadas de datos.

En nuestro caso no será necesario actualizar los datos antiguos, ya que no se han definido diferentes grupos a los que puedan pertenecer los datos.

8. Si se aplicó la política pero está deshabilitada, debe activarse la política Oracle Label Security.

La política ya se encuentra activada, por lo que este punto no es necesario llevarlo a cabo.

Comprobación del funcionamiento de la política Oracle Label Security

Una vez que se ha creado la política Oracle Label Security *"PROTECT_PII"* y la política VPD asociada *"VPD_PROTECT_PII"* se procede a comprobar su funcionamiento a través del siguiente ejemplo.

- Acceder a la base de datos como el usuario PRUEBA2 (personal no perteneciente a recursos humanos) y realizar una consulta sobre la tabla de datos *"hr_app.nominas"*.

Se deberá acceder a SQL*PLUS con el usuario *"HR_APP"* y lanzar el siguiente script:

```
exec dbms_application_info.set_client_info('PRUEBA2')
exec sa_session.set_access_profile('PROTECT_PII',sys_context('userenv','client_info'))
select sys_context('userenv','session_user') as "Session user",
     sa_session.sa_user_name('PROTECT_PII') as "Clearance of" from dual
/
select * from hr_app.nominas where rownum <6
/
```

El resultado que se obtendrá es el que se muestra a continuación:

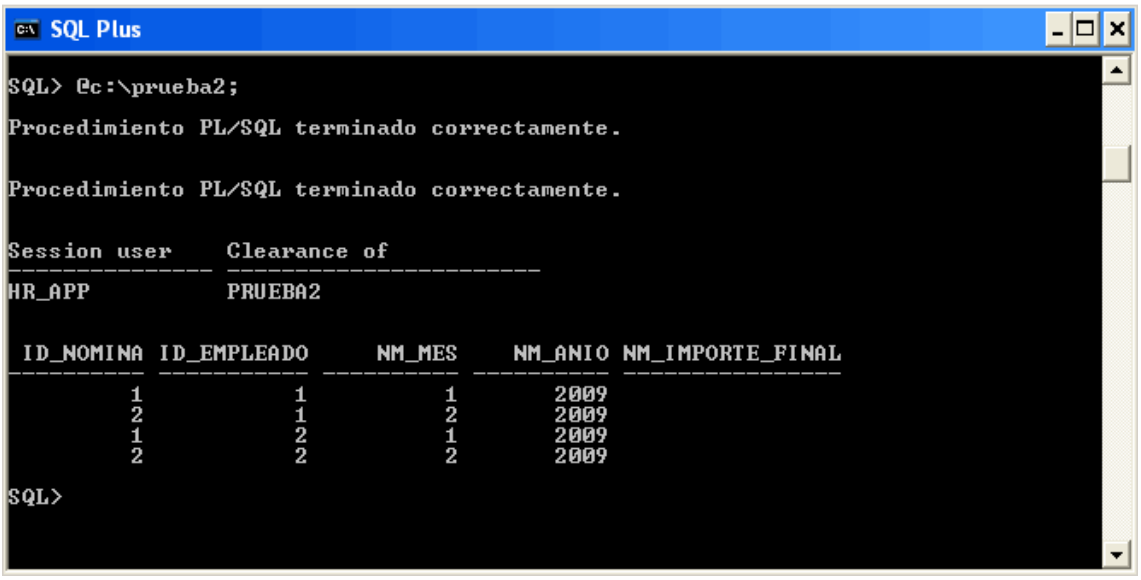

Imagen 73. Datos devueltos al realizar la consulta con el usuario *"Prueba2"*

Como puede observarse el campo "*nm_importe_final*" aparece con el valor nulo.

- Acceder a la base de datos como el usuario PRUEBA (personal perteneciente a recursos humanos) y realizar una consulta sobre la tabla de datos *"hr_app.nominas"*.

Se deberá acceder a SQL*PLUS con el usuario *"HR_APP"* y lanzar el siguiente script:

```
exec dbms_application_info.set_client_info('PRUEBA')
exec sa_session.set_access_profile('PROTECT_PII',sys_context('userenv','client_info'))
select sys_context('userenv','session_user') as "Session user",
      sa_session.sa_user_name('PROTECT_PII') as "Clearance of" from dual
/
select * from hr_app.nominas where rownum <6

/
```

El resultado que se obtendrá es el que se muestra a continuación:

Imagen 74. Datos devueltos al realizar la consulta con el usuario *"Prueba"*

Capítulo 7: Uso de Oracle Label Security para gestionar el etiquetado de datos

Introducción

Hasta ahora, se han explicado las nociones básicas para la implementación de una política Oracle Label Security sin que esta estuviera aplicada sobre una tabla de base de datos.

Para implementar una política Oracle Label Security aplicada sobre una tabla de base de datos, es necesario comprender el proceso a seguir para etiquetar los datos de dicha tabla.

Antes de proceder con la implementación de una política aplicada sobre una tabla de base de datos, el analista de la aplicación tendrá que haber realizado los pasos previos de análisis, expuestos en el capítulo anterior, con el fin de saber las diferentes etiquetas de datos que van a representar los datos.

Política Oracle Label Security aplicada sobre una tabla de base de datos

Cuando una política se aplica sobre una tabla de base de datos se creará una columna en la base de datos. El valor de esta columna será el que proporcionará a Oracle Label Security la información necesaria para saber si un usuario puede acceder a ver o modificar su contenido. De forma predeterminada, el tipo de datos de la columna correspondiente a la etiqueta es de tipo *number*.

Cada fila de datos de la tabla deberá tener un valor numérico (*label tag*) en la columna asociada a la política Oracle Label Security. El valor equivalente corresponderá con una etiqueta de datos creada anteriormente en la política.

Ocultación de la columna asociada a la política Oracle Label Security

El administrador podrá decidir no mostrar la columna que representa la política mediante la opción de ocultación en la tabla. Después de ocultar la política aplicada sobre una tabla, un usuario que ejecute la sentencia *select* o *describe* no verá la columna asociada a la política. Si la columna asociada a la política no se oculta, entonces esta se muestra.

Label Tag

Al crear etiquetas, el administrador especificará el conjunto de combinaciones válidas de los componentes que podrán constituir una etiqueta, es decir, un nivel combinado opcionalmente con uno o más grupos o compartimentos. Cada una de esas etiquetas dentro de la política será únicamente identificada por un código numérico asociado. Este código numérico será asignado por el administrador o generado automáticamente a partir de su primera utilización. La definición manual proporcionará al administrador la ventaja de controlar el orden de los valores de la etiqueta cuando sean ordenados o comparados lógicamente.

Sin embargo, el label tag debe ser único en todas las políticas de base de datos. Cuando se usan múltiples políticas en la base de datos, no se puede utilizar el mismo label tag en las diferentes políticas.

En la columna que representa la política dentro de la tabla de base de datos se almacena el label tag asociado a la etiqueta, no la cadena de caracteres que representa a la etiqueta.

A continuación, se presentan las diferentes formas en las que se puede definir una etiqueta.

Definición manual de label tags para la ordenación de etiquetas

El administrador puede aplicar una estrategia de manipulación de datos que permita a las etiquetas ser ordenadas y puedan compararse de manera lógica. Anteriormente, deberá haber predefinido todas las etiquetas asociadas a la política y asignar a cada etiqueta un label tag.

Puede ser ventajoso aplicar una estrategia en la que el label tag tenga un valor en el que cada dígito represente un componente de la etiqueta. Además, asociada con esta estrategia, puede ser una buena práctica asignar valores más altos según el nivel de sensibilidad de la etiqueta.

Definición manual de label tags para la manipulación de datos

El label tag puede utilizarse como una manera conveniente de partición de datos. Por ejemplo, se podría particionar una tabla de datos de forma que aquellos datos que tengan asociada una etiqueta de datos en el rango 1000-1999 sean almacenados en la primera partición de la tabla, los datos con etiquetas en el rango 2000-2999 sean almacenados en la segunda partición, y así sucesivamente.

Esta simplificación en la notación también puede ser útil cuando hay un número finito de etiquetas y se necesita realizar diversas operaciones sobre ellas. Por ejemplo, una empresa alberga un sistema de recursos humanos para muchas otras empresas. Los usuarios de la empresa *Y* tienen la etiqueta *"C:ALPHA:CY"*, para la que se ha establecido el label tag 210. Para determinar el número total de usuarios de la empresa *Y*, solamente debería ejecutarse la siguiente sentencia:

*Select * from <tabla> where <columna_etiqueta> = 210;*

Generación automática de label tags

Mediante la generación automática de label tag, los dígitos que lo componen no tienen ninguna relación con los números asignados a cada componente de la etiqueta asociada. En consecuencia, no habrá ninguna manera de agrupar los datos.

Representación de la etiqueta de datos

Como se ha expuesto con anterioridad, el valor que se almacena en el campo asociado a la política dentro de una tabla de base de datos corresponde a un número. Al recuperarse el valor de este campo, no se obtiene automáticamente la cadena de caracteres que representa a la etiqueta asociada a dicho valor.

Existen dos funciones que permiten la manipulación de etiquetas.

Char_to_label

Permite la conversión de la cadena de caracteres que representa la etiqueta a su label tag asociado.

Sintaxis:

Function char_to_label (<nombre_política> in varchar2,

<etiqueta_cadena_caracteres> in varchar2)

Return number;

Label_to_char

Permite la conversión del label tag a la cadena de caracteres que representa a la etiqueta asociada.

Sintaxis:

Function label_to_char (<label_tag> in number) Return vachar2;

Filtrado de datos mediante etiquetas

Las etiquetas de datos asociadas a una tabla se pueden utilizar para filtrar datos a la hora de realizar consultas, dentro de aquellos datos que el usuario está autorizado a ver. A la hora de realizar inserción, modificación o borrado de datos, Oracle Label Security permite o deniega la operación solicitada, en base a la autorización que tenga el usuario.

Utilización del label tag asociado a una etiqueta en cláusulas "where"

Al utilizar las etiquetas en el formato numérico, se pueden crear cláusulas *"where"* mediante operadores lógicos.

Por ejemplo, si ha asignado a las etiquetas *"SIN CLASIFICAR"* el rango 1000-1999, a las etiquetas *"SENSIBLE"* el rango 2000-2999, y a las etiquetas *"MUY SENSIBLE"* el rango 3000 en adelante. Se puede listar todos los registros con etiqueta asociada *"SENSIBLE"* a través de la siguiente consulta:

*Select * from <tabla> where <columna_etiqueta> between 2000 and 2999;*

Ordenación de datos mediante el valor del label tag

Se puede ordenar las filas que se han obtenido como resultado de la consulta mediante el valor numérico de la etiqueta asociada a la fila de datos. Por ejemplo:

*Select * from <tabla> order by <columna_etiqueta>;*

Ordenación de datos mediante la cadena de caracteres que representa a la etiqueta asociada al label tag

Usando la función *label_to_char*, se pueden ordenar las filas que se han obtenido como resultado de la consulta mediante la cadena de caracteres que representa a la etiqueta. Por ejemplo:

*Select * from <tabla> order by label_to_char(<columna_etiqueta>);*

Determinación de los límites superior e inferior de las etiquetas

Oracle Label Security dispone de funciones para determinar el menor límite superior o el mayor límite inferior entre dos o más etiquetas.

Least_ubound

La función *least_ubound* devuelve la cadena de caracteres que representa la etiqueta que es el menor límite superior entre la etiqueta_1 y la etiqueta_2, es decir, una etiqueta que contiene a ambas.

Least_ubound = Mayor nivel + Unión de compartimentos + Unión de grupo

Sintaxis:

Function Least_ubound (<etiqueta_1> in number,

<etiqueta_2> in number) Return vachar2;

Esta función es útil cuando se realiza la unión de filas con diferentes etiquetas, ya que proporciona una etiqueta que hace de marca de agua superior para la unión de filas.

Greatest_lbound

La función *greatest_lbound* devuelve la cadena de caracteres que representa la etiqueta que es el mayor límite inferior entre la etiqueta_1 y la etiqueta_2.

Greatest_lbound = Menor nivel + Intersección de compartimentos + Intersección de grupos

Sintaxis:

Function Greatest_ubound (<etiqueta_1> in number,

<etiqueta_2> in number) Return vachar2;

Etiquetado de datos

Al realizar la inserción de datos en una tabla protegida por una política Oracle Label Security, el valor del label tag debe ser suministrado, generalmente, en la propia sentencia *insert*.

Para ello, debe especificar explícitamente el label tag o convertir la cadena de caracteres que representa la etiqueta al label tag asociado.

Las únicas veces que en la inserción de datos se puede omitir el valor de la etiqueta son:

- si cuando se aplicó la política se especificó la opción *LABEL_DEFAULT*.
- si se ha asociado una función de etiquetado a la política.

A continuación, se muestran las diferentes formas de asignar un valor al campo asociado a la política Oracle Label Security.

Inserción de etiquetas usando la función char_to_label

A la hora de insertar los datos, se puede especificar la cadena de caracteres que representa la etiqueta a asignar transformándola mediante el uso de la función *char_to_label*.

Inserción de etiquetas usando el valor numérico del label tag

Se pueden insertar los datos, utilizando el label tag asociado a una etiqueta.

Introducción de datos sin especificar el valor de la etiqueta

Si la opción *LABEL_DEFAULT* está activada, o hay una función de etiquetado sobre la tabla en la que se aplica la política, no será necesario especificar la etiqueta a la hora de insertar los datos.

Inserción de etiquetas usando la función to_data_label

Si se están generando nuevas etiquetas dinámicamente, entonces a la hora de insertar los datos se puede utilizar la función *to_data_label*, la cual creará automáticamente una etiqueta de datos válidos.

Para poder utilizar esta opción, el usuario debe tener permiso de ejecución sobre la función *to_data_label.*

Sintaxis:

Function to_data_label (<nombre_politica> in varchar2,

<nombre_etiqueta> in varchar2) Return number;

Caso práctico

Una empresa dispone de una tabla de base de datos con todos los datos de sus empleados de España. Con el fin de optimizar recursos y aumentar el nivel de seguridad, han decidido que se implemente una política Oracle Label Security sobre la tabla.

Esta política deberá recoger los requisitos funcionales que se describen a continuación:

- *Según el departamento o área al que pertenezca un empleado, sus datos serán confidenciales, sensibles o muy sensibles:*

  Informática: Confidencial

  Recursos Humanos: Sensible

  Dirección: Muy Sensible

- *Por cada uno de los departamentos anteriormente expuestos, se creará un compartimento.*

- *La empresa dispone de tres sedes: Asturias, Barcelona y Madrid, con el fin de segmentar la información en grupos, se creará un grupo por cada sede.*

Los usuarios autorizados a consultar la tabla de empleados deben ser:

- *Empleados del departamento de informática. Sólo podrán consultar los datos de aquellos empleados que pertenezcan a su mismo departamento y sede.*

  INFO_ASTURIAS: Usuario genérico del departamento de informática y sede en Asturias.

  INFO_BARCELONA: Usuario genérico del departamento de informática y sede en Barcelona.

  INFO_MADRID: Usuario genérico del departamento de informática y sede en Madrid.

- *Empleados del departamento de recursos humanos. Podrán consultar los datos de aquellos empleados pertenecientes al departamento de informática y recursos humanos, y cuya sede sea la misma.*

  RRHH_ASTURIAS: Usuario genérico del departamento de recursos humanos y sede en Asturias.

  RRHH_BARCELONA: Usuario genérico del departamento de recursos humanos y sede en Barcelona.

  RRHH_MADRID: Usuario genérico del departamento de recursos humanos y sede en Madrid.

- *Empleados del área de dirección. Podrá consultar los datos de todos los empleados de la empresa.*

  DIRECCION: Usuario genérico del área de dirección.

- *Administrador de la política EJEMPLOS_PFC.*

1. Crear la política Oracle Label Security.

Accediendo a Oracle Enterprise Manager, se procederá a crear la política *"Ejemplo_1"*. Política encargada de dar solución al problema anteriormente expuesto.

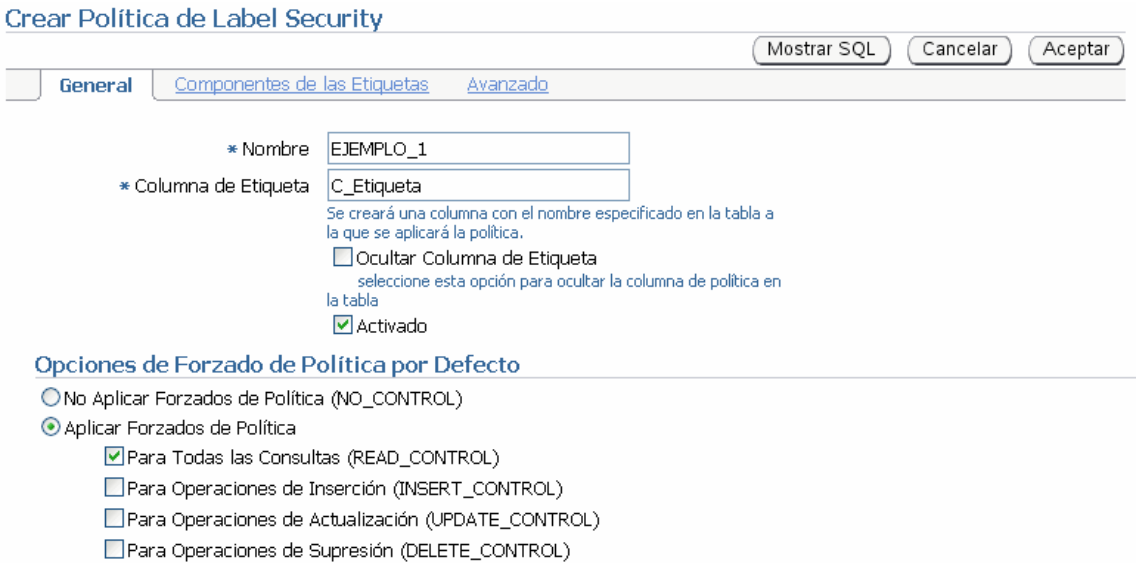

Imagen 75. Oracle Enterprise Manager. Creación de la política *"Ejemplo_1"*

Como puede observarse en la imagen, la columna asociada a la etiqueta *"C_Etiqueta"* será la columna que se creará en la tabla de base de datos sobre la que se va a aplicar la política.

Además, se ha seleccionado la opción de aplicar forzados de políticas para todas las consultas, es decir, se deberá aplicar la política sobre todas las consultas que se realicen.

2. Definir todos los componentes para etiquetas de datos válidos utilizando niveles, compartimentos y grupos con la información recopilada durante el análisis.

Se procederá a crear los tres niveles de confidencialidad de información, asociándole a cada uno una etiqueta, junto con los compartimentos y grupos expuestos en el enunciado.

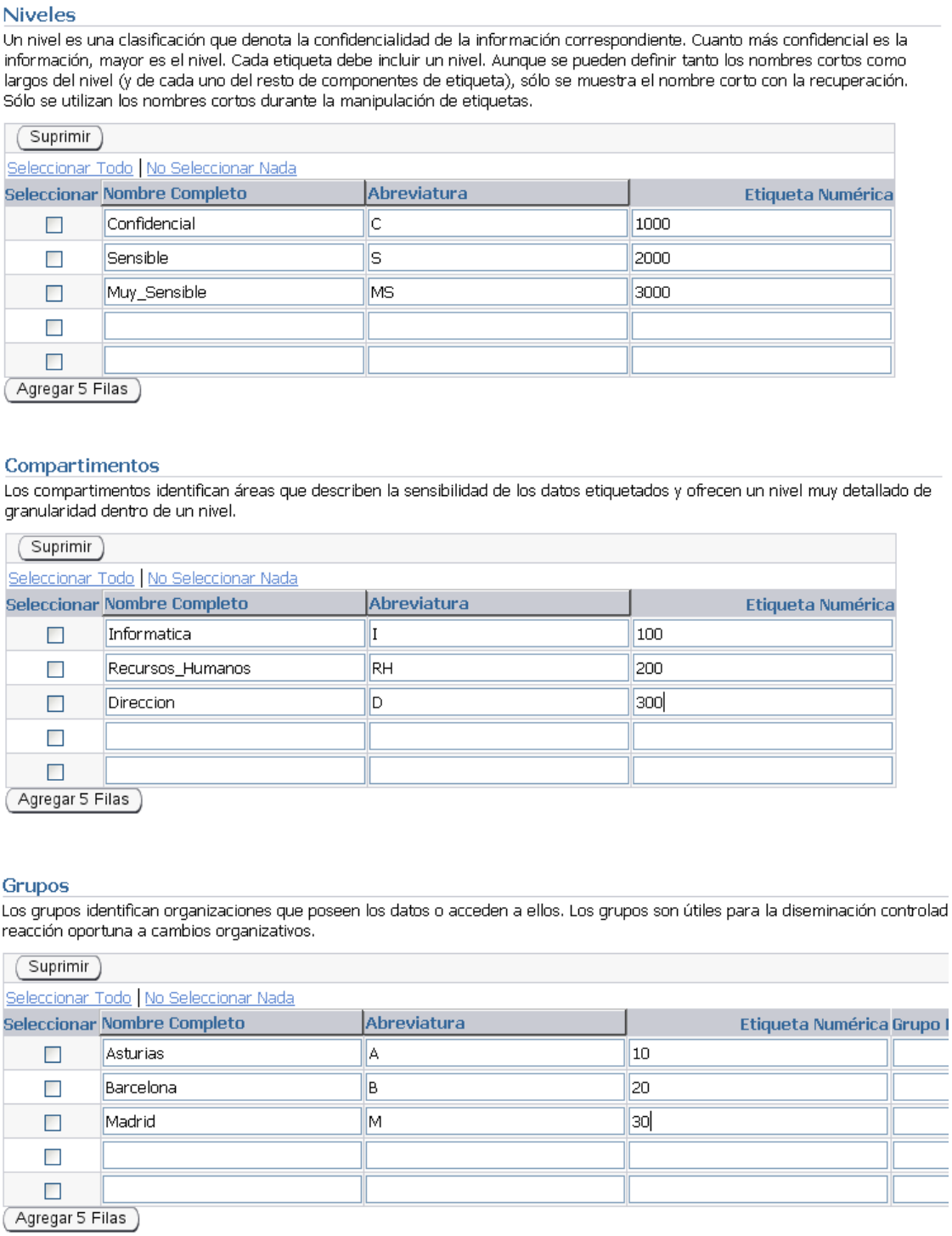

Imagen 76. Oracle Enterprise Manager. Resumen definición componentes etiqueta

A la hora de definir la etiqueta numérica asociada al nivel y compartimento, se ha llevado a cabo la estrategia de asociar un valor numérico mayor según aumenta la sensibilidad de los mismos.

3. Asociación de etiquetas de datos a la política Oracle Label Security.

Antes de generar las etiquetas de datos asociadas a la política, el administrador deberá haber realizado un análisis exhaustivo de los requisitos funcionales. En esta ocasión, a partir del enunciado se puede extraer lo siguiente:

Se necesitará una etiqueta de datos para los empleados que pertenezcan al departamento de informática, teniendo en cuenta la sede a la que pertenecen. Como se expone en el enunciado, los datos asociados a este tipo de empleados deben tener un nivel de sensibilidad confidencial.

Así mismo, se necesitará una etiqueta de datos para los empleados que pertenezcan al departamento de recursos humanos, teniendo en cuenta la sede a la que pertenecen y permitiendo. El nivel de sensibilidad de sus datos es sensible.

Por último, se necesitará una etiqueta de datos para los empleados pertenecientes al área de dirección. El nivel de sensibilidad de sus datos es muy sensible.

A continuación se muestra una tabla resumen:

Etiqueta	Valor Numérico	Descripción
C:I:A	1110	Informáticos pertenecientes a la sede de Asturias.
C:I:B	1120	Informáticos pertenecientes a la sede de Barcelona
C:I:M	1130	Informáticos pertenecientes a la sede de Madrid.
S:RH:A	2210	Personal de Recursos Humanos pertenecientes a Asturias.
S:RH:B	2220	Personal de Recursos Humanos pertenecientes a Barcelona.
S:RH:C	2230	Personal de Recursos Humanos pertenecientes a Madrid.
MS:D	3300	Personal perteneciente al área de dirección.

Tabla 9. Tabla resumen de etiquetas de la política OLS *"Ejemplo_1"*

Nota: A la hora de asignar el valor numérico asociado a una etiqueta se ha utilizado la definición manual de label tags, para que estas puedan ordenarse y manipularse fácilmente.

Para proceder a dar de alta las etiquetas de datos, habrá que marcar la política deseada, en el desplegable de acciones seleccionar "Etiquetas de Datos" y pulsar el botón "Ir".

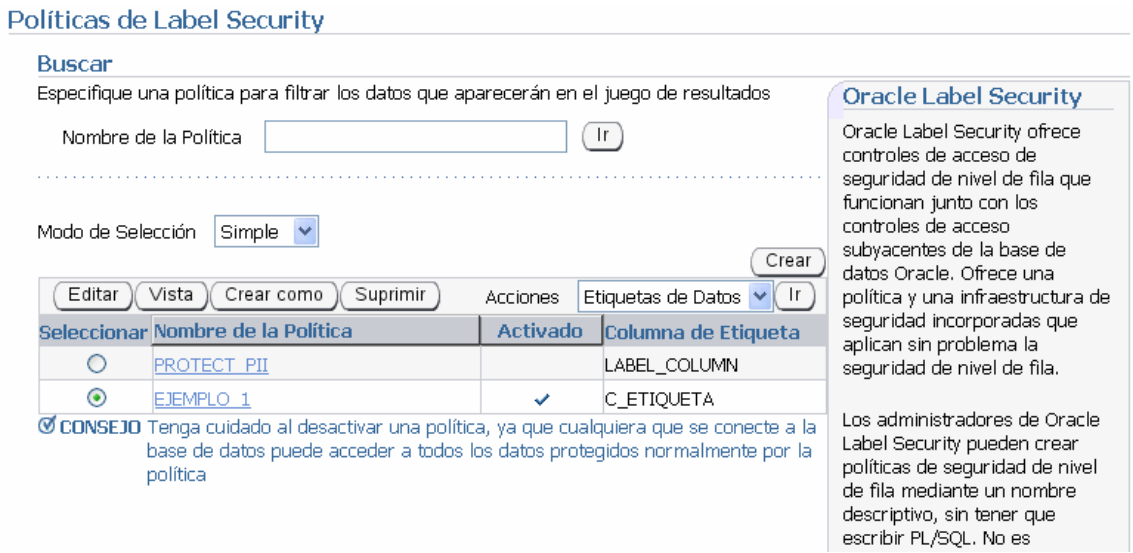

Imagen 77. Oracle Enterprise Manager. Selección de política a modificar etiquetas de datos

En la pantalla "Etiquetas de Datos", se tendrá que pulsar el botón "Agregar" para comenzar con la creación de una etiqueta de datos.

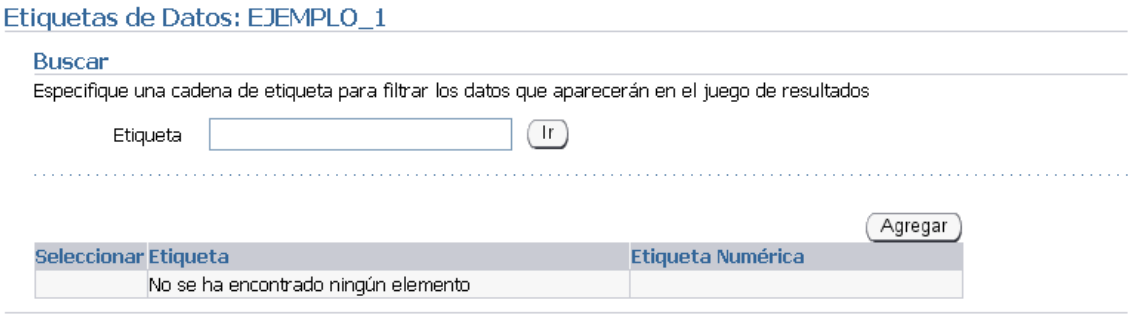

Imagen 78. Oracle Enterprise Manager. Agregar etiqueta de datos

A continuación, se muestran los pasos seguidos para crear la etiqueta de datos *"C:I:A"*.

Inicialmente, se especificará el código numérico de la etiqueta y el nivel.

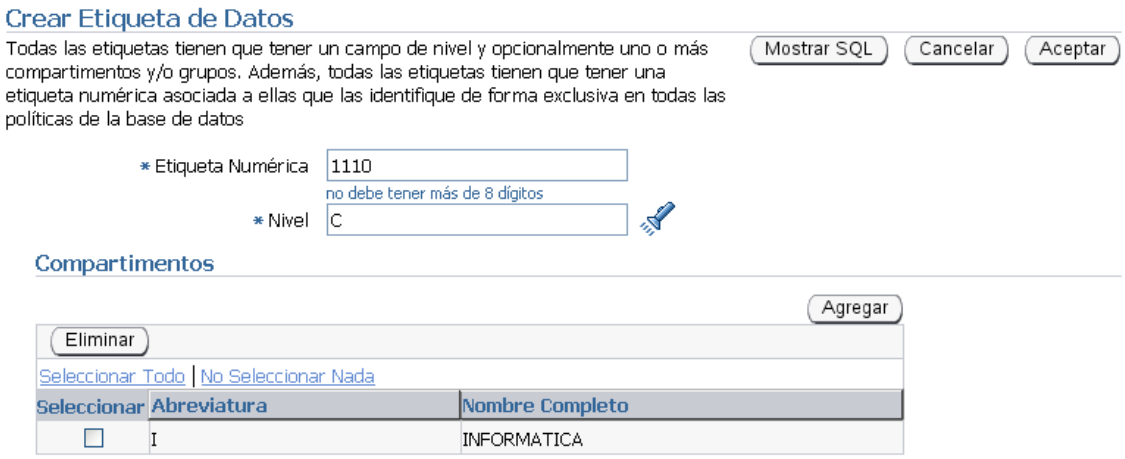

Imagen 79. Oracle Enterprise Manager. Creación de la etiqueta de datos *"C:I:A" (Paso 1)*

Seguidamente, se agregarán los compartimentos y grupos asociados a la etiqueta.

Imagen 80. Oracle Enterprise Manager. Creación de la etiqueta de datos *"C:I:A" (Paso 2)*

Finalmente, pulsando el botón "Aceptar" se creará la etiqueta de datos. Estos pasos deberán repetirse para la creación de cada una de las etiquetas restantes.

Una vez creadas todas las etiquetas de datos, en la pantalla "Etiqueta de Datos" se visualizará el resumen de etiquetas de datos asociadas a la política Oracle Label Security.

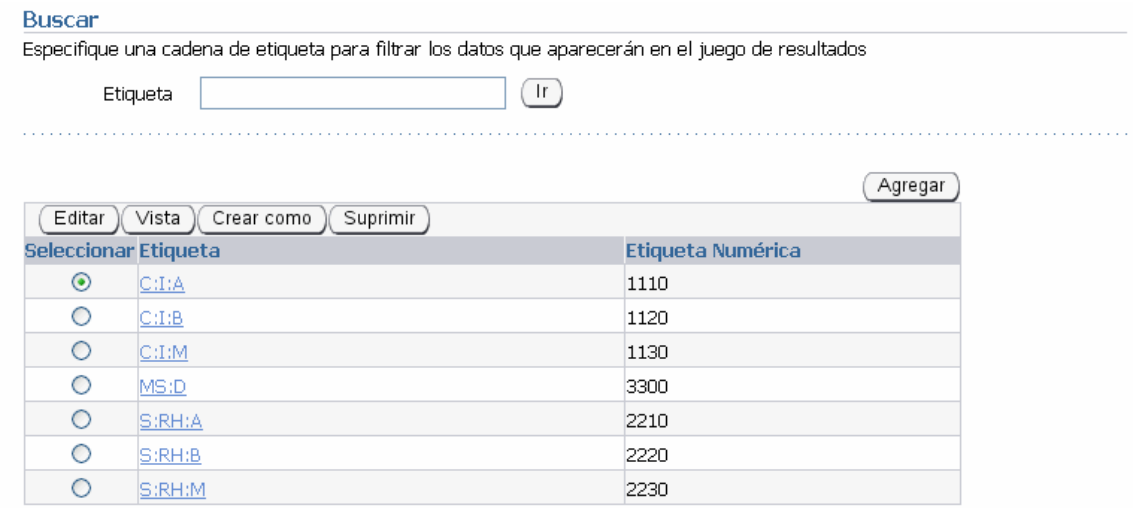

Imagen 81. Oracle Enterprise Manager. Resumen etiquetas de datos asociadas a la política *"Ejemplo_1"*

4. Aplicar la política Oracle Label Security sobre la tabla de base de datos.

Para asignar a la política Oracle Label Security la tabla (o esquema de base de datos) sobre la que se va a aplicar, habrá que marcar la política deseada, a continuación, en el desplegable seleccionar la opción "Aplicar" y pulsar el botón "Ir".

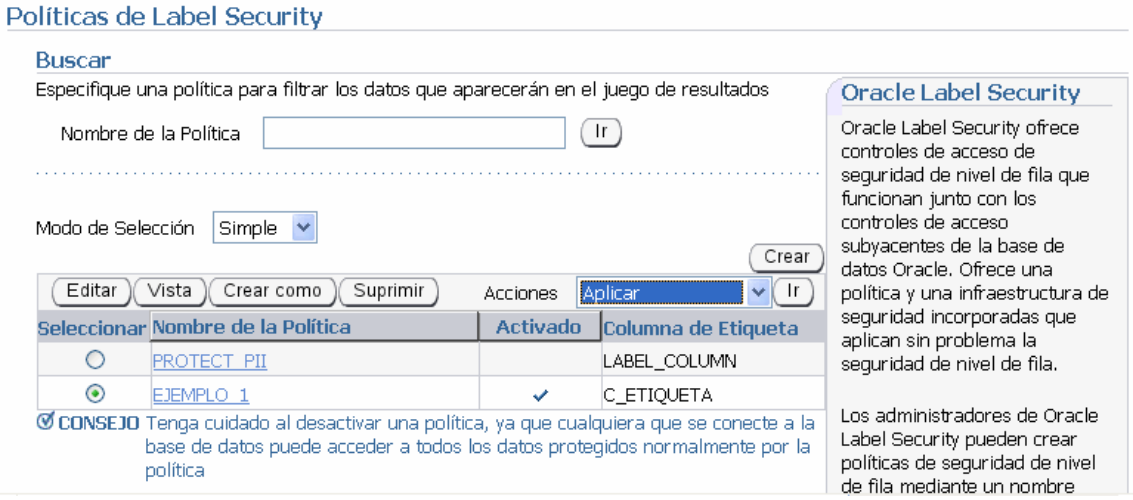

Imagen 82. Oracle Enterprise Manager. Acción aplicar la política *"Ejemplo_1"*

En la ventana "Agregar Tabla", se seleccionará la tabla que se va a asociar a la política Oracle Label Security, así como la opción de ocultación de la columna de la política, activar la política sobre la tabla y las opciones de forzado de la política.

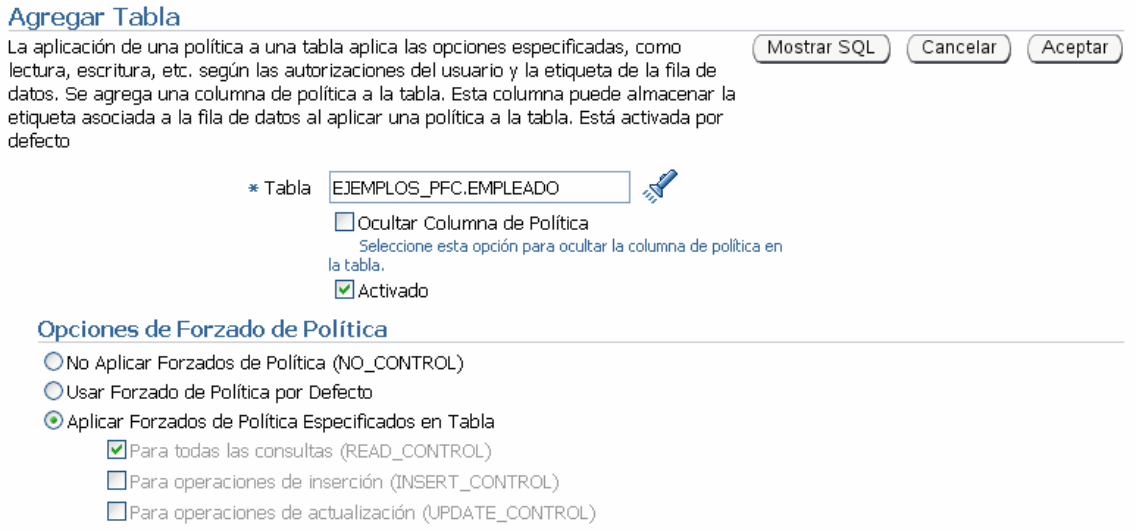

Imagen 83. Oracle Enterprise Manager. Agregar tabla *"Ejemplos_pfc.empleado"*

Como puede observarse, en este ejemplo, se ha activado la política Oracle Label Security para la tabla *ejemplos_pfc.empleado* y se aplicará el forzado de la política para todas las consultas de datos.

5. Autorizar a los usuarios.

Siguiendo las especificaciones dadas en el enunciado, se ha procedido a crear los diferentes usuarios con sus correspondientes permisos.

A continuación se muestra, cómo quedaría la ventana resumen de usuarios autorizados en la política.

Agregar Usuarios

Editar | Vista | Crear como | Suprimir

Seleccionar	Usuario	Etiqueta de Lectura Máxima	Etiqueta de Escritura Máxima	Privilegios
◉	INFO_ASTURIAS	C:I:A	C	
○	INFO_BARCELONA	C:I:B	C	
○	INFO_MADRID	C:I:M	C	
○	RRHH_ASTURIAS	S:I,RH:A	S	
○	RRHH_BARCELONA	S:I,RH:B	S	
○	RRHH_MADRID	S:I,RH:M	S	
○	DIRECCION	MS:I,RH,D:A,B,M	MS	
○	EJEMPLOS_PFC			Profile Access

Imagen 84. Oracle Enterprise Manager. Tabla resumen de usuarios autorizados en la política *"Ejemplo_1"*

6. Pruebas realizadas.

En la inserción de datos en la tabla *ejemplos_pfc.empleado* para asignar un valor a la columna asociada a la política Oracle Label Security, se ha utilizado la función *char_to_label*.

Ejemplo:

```
Insert into ejemplos_pfc.empleado
        (c_id, c_nif, ds_nombre, ds_apell1, ds_apell2, ds_tlfno, c_deptno, c_etiqueta)
values  ('9', '00000009H' ,'Felipe', 'Gallego', 'Gil', '986521498', 'INF',
        char_to_label('EJEMPLO_1','C:I:A');
```

Donde:

C_id: Identificador interno del empleado (Primary Key).

C_nif: Código alfanumérico de identificación del empleado.

Ds_nombre: Nombre del empleado.

Ds_apell1 y Ds_apell2: Primer y segundo apellido del empleado respectivamente.

Ds_tlfno: Teléfono del empleado.

C_deptno: Código del departamento al que pertenece el empleado.

C_etiqueta: Campo asociado a la política Oracle Label Security.

Una vez insertados los datos en la tabla *ejemplos_pfc.empleado*, para comprobar el buen funcionamiento de la política Oracle Label Security asociada a la tabla, se ha ejecutado el siguiente script en SQL Plus, accediendo a la base de datos con el usuario *ejemplos_pfc*.

```
col "Conexion:" format a50
exec dbms_application_info.set_client_info(<usuario>)
exec sa_session.set_access_profile('EJEMPLO_1',sys_context('userenv','client_info'))
select 'Usuario '||sys_context('userenv','session_user')||' conectado como '||
    sa_session.sa_user_name('EJEMPLO_1') as "Conexion:" from dual
/
col "Nombre" format a25
col "Etiqueta" format a15
Select e.c_nif as "DNI",
    e.ds_nombre||' '||e.ds_apell1||' '||e.ds_apell2 as "Nombre",
    e.ds_tlfno as "Tlfno",
    e.c_deptno as "Deptno",
    label_to_char(c_etiqueta) as "Etiqueta"
    from ejemplos_pfc.empleado e
/
```

Donde **<*usuario*>** es el usuario autorizado con el que se quiere acceder a la política Oracle Label Security.

A continuación, se muestran los resultados obtenidos según el usuario con el que se acceda a la política:

<usuario> = 'DIRECCION'

Con el fin de mostrar todos los empleados dados de alta en la tabla *ejemplos_pfc.empleado*, inicialmente, se muestra el resultado obtenido accediendo en la política Oracle Label Security con el usuario *DIRECCION*.

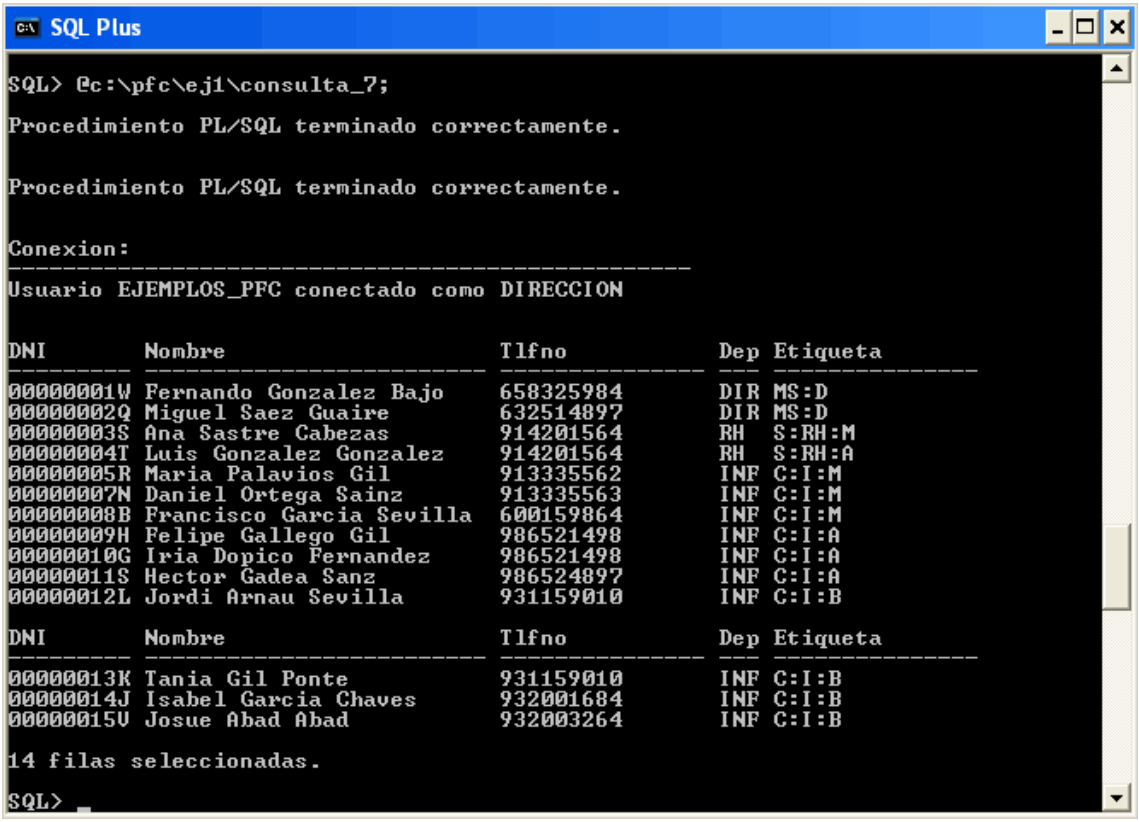

Imagen 85. Resultado obtenido con el usuario *DIRECCION*

A través de la etiqueta asociada que tiene cada empleado puede saberse el departamento y delegación a la que pertenece, siendo:

Departamentos / Áreas

I: Dpto. de Informática	RH: Dpto. de Recursos Humanos	D: Área de Dirección

Delegaciones

A: Asturias	B: Barcelona	M: Madrid

<usuario> = 'INFO_ASTURIAS'

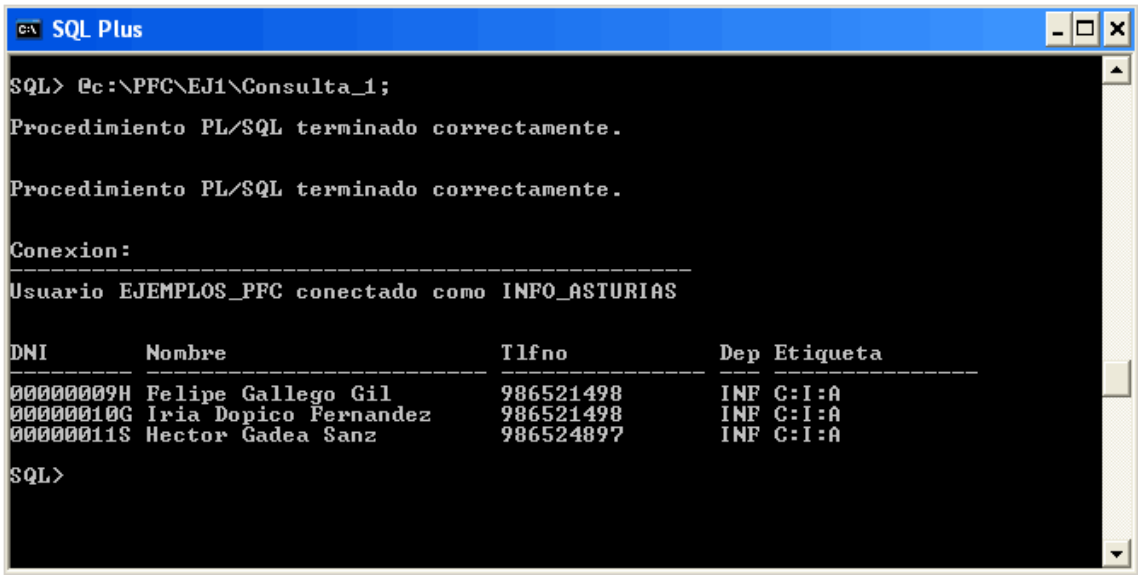

Imagen 86. Resultado obtenido con el usuario *INFO_ASTURIAS*

<usuario> = 'RRHH_ASTURIAS'

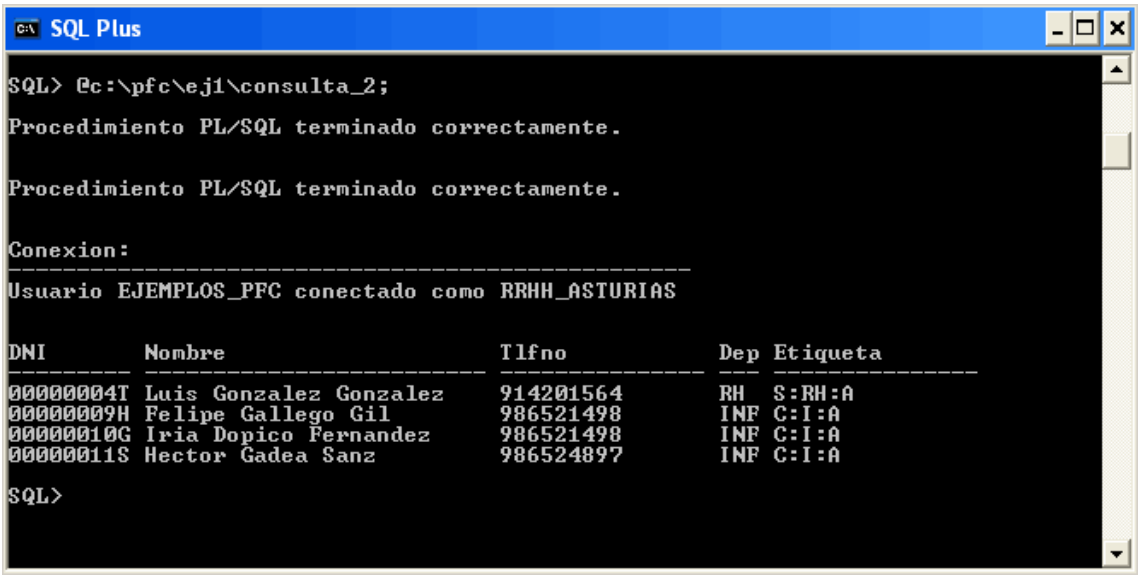

Imagen 87. Resultado obtenido con el usuario *RRHH_ASTURIAS*

<usuario> = 'INFO_BARCELONA'

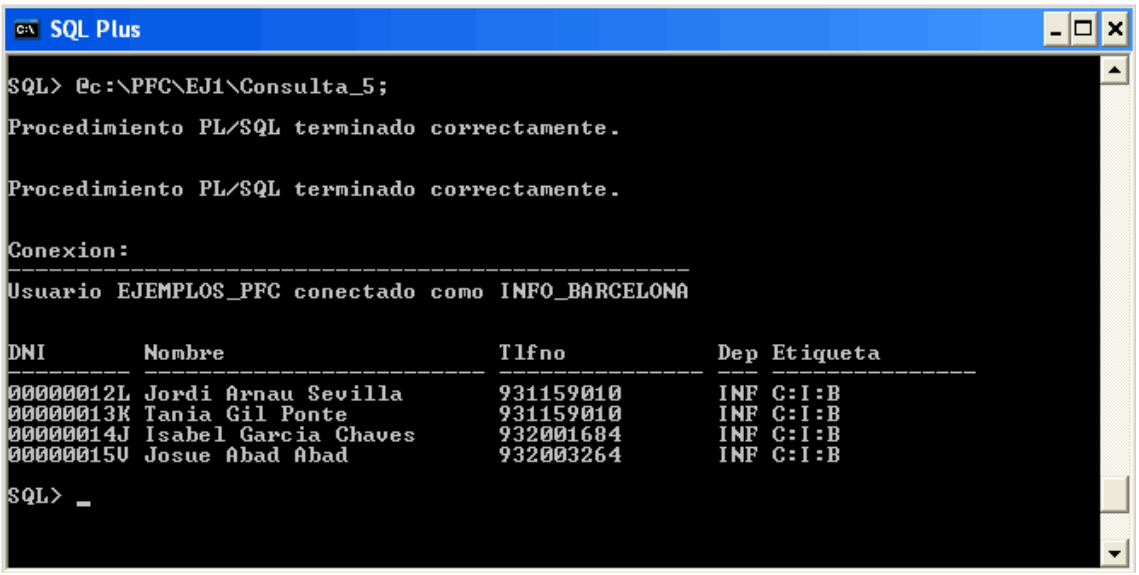

Imagen 88. Resultado obtenido con el usuario *INFO_BARCELONA*

<usuario> = 'RRHH_BARCELONA'

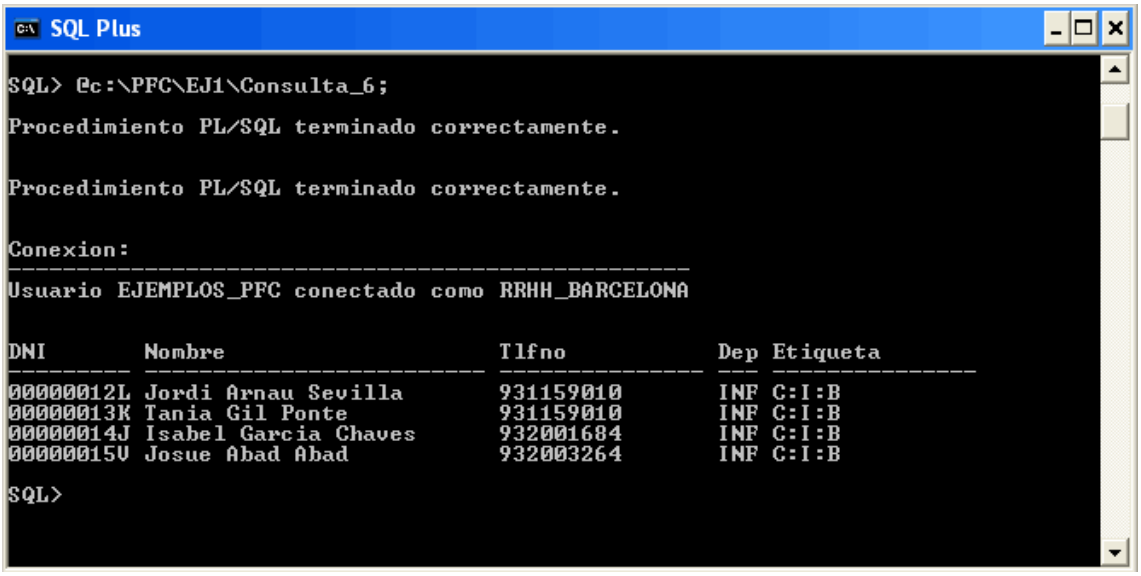

Imagen 89. Resultado obtenido con el usuario *RRHH_BARCELONA*

<usuario> = 'INFO_MADRID'

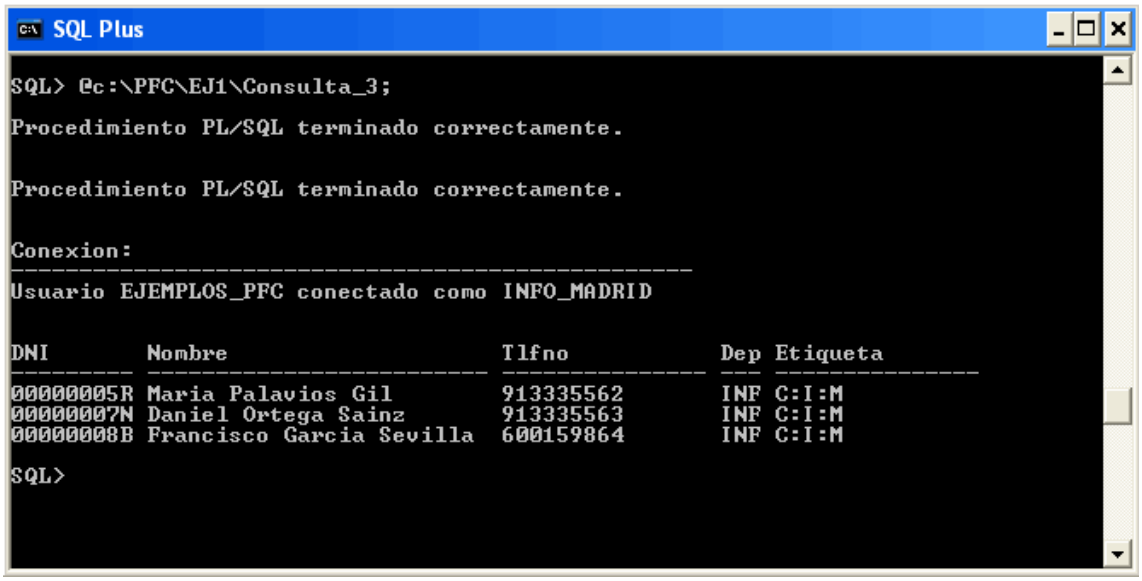

Imagen 90. Resultado obtenido con el usuario *INFO_MADRID*

<usuario> = 'RRHH_MADRID'

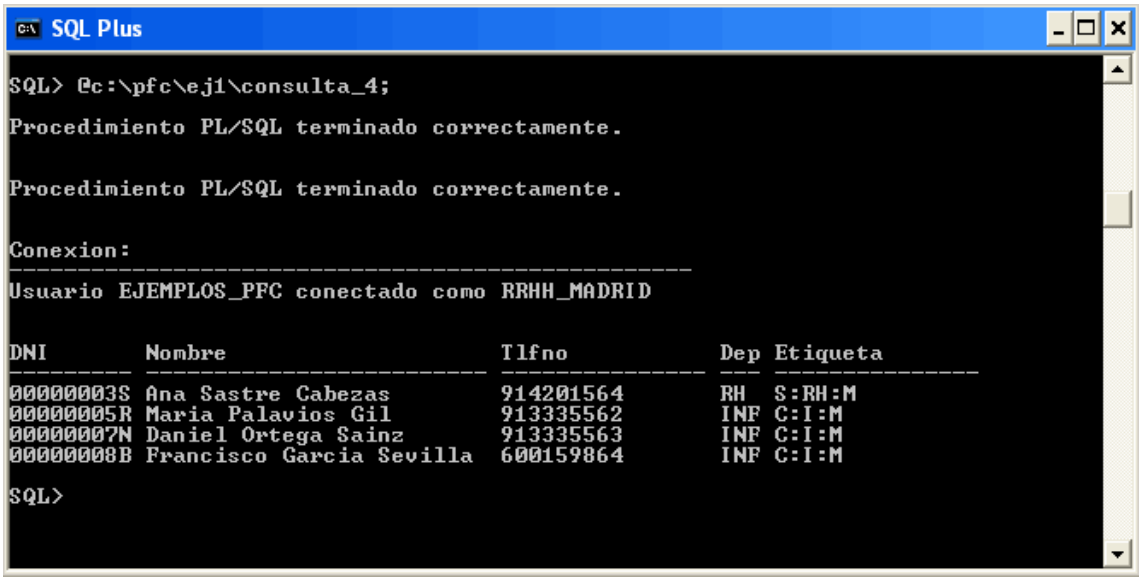

Imagen 91. Resultado obtenido con el usuario *RRHH_MADRID*

Capítulo 8: Auditoría en Oracle Label Security

Introducción

La auditoría de Oracle Database 11g, permite mantener una pista en la base de datos acerca de la responsabilidad de los usuarios sobre cada una de las acciones que realiza. Puede realizarse un seguimiento específico sobre los objetos de la base de datos, operaciones, usuarios y privilegios.

Oracle Label Security completa este seguimiento mediante el uso de su propia gestión administrativa y la política de privilegios. Proporciona el paquete *SA_AUDIT_ADMIN* para establecer y modificar las opciones de auditoría sobre una política Oracle Label Security.

Panorama general de la auditoría de Oracle Label Security

Como se ha mencionado anteriormente, Oracle Label Security completa el estándar de auditoría de la base de datos Oracle, mediante el uso de su propia gestión administrativa y la política de privilegios. Para ello puede utilizarse el paquete de procedimientos *SA_AUDIT_ADMIN* u, opcionalmente, *Oracle Enterprise Manager* con el fin configurar y cambiar las opciones de auditoría de una política Oracle Label Security.

Al crearse una nueva política, la columna de etiqueta de la política se añade en una pista de auditoría en la base de datos. Esta se crea con independencia de si la auditoría está activada o no. Siempre que un registro se introduce en la tabla de auditoría, cada una de las políticas proporciona una etiqueta para el registro que indica la sesión de etiqueta. El administrador puede crear vistas sobre la auditoría para mostrar estas etiquetas.

Las opciones de auditoría que se especifiquen se aplicarán a los períodos de sesiones posteriores. Se puede especificar opciones de auditoría, incluso si la auditoría está desactivada. Cuando se habilita la auditoría de Oracle Label Security, las opciones entran en vigor.

Debe tenerse en cuenta que Oracle Label Security no proporciona etiquetas para los datos de auditoría que se registran en la pista de auditoría del sistema operativo. Todos los registros de auditoría de Oracle Label Security se registran directamente en la pista de auditoría de la base de datos. Si la auditoría está desactivada, entonces Oracle Label Security no genera registros de auditoría.

Habilitar el sistema de auditoría: Parámetro de inicialización *AUDIT_TRAIL*

Para que Oracle Label Security genere registros de auditoría, primero debe habilitarse el sistema de auditoría mediante el establecimiento del parámetro de inicialización *AUDIT_TRAIL* en el fichero de parámetros de la base de datos.

El parámetro puede tomar uno de los siguientes valores:

DB: Habilita la auditoría de la base de datos y dirige todas las pistas de auditoría a la tabla *AUD$* de la base de datos. Valor recomendado por Oracle.

DB_EXTENDED: Recoge la funcionalidad asociada al parámetro *AUDIT_TRAIL = DB* y también rellena las columnas *SqlBind* y *SqlText* de la tabla *AUD$.*

OS: Habilita la auditoría del sistema operativo. Dirige la mayor parte de pistas de auditoría de la base de datos Oracle al sistema operativo, en lugar de a la base de datos.

Si se establece *AUDIT_TRAIL = OS*, las pistas de auditoría específicas de Oracle Label Security se almacenarán en la tabla *AUD$* de la base de datos y el resto de pistas de auditoría de Oracle se almacenarán en la auditoría del sistema operativo.

NONE: Desactiva la auditoría. Valor por defecto.

Una vez editado el parámetro del fichero, debe reiniciarse la instancia de base de datos para activar o desactivar la auditoría de Oracle, según el valor del parámetro *AUDIT_TRAIL*.

Se debe configurar el parámetro *AUDIT_TRAIL* antes que las opciones de auditoría. En caso contrario, aunque se configuren las opciones de auditoría, las pistas de auditoría no se almacenarán en la base de datos.

Habilitar la auditoría de Oracle Label Security con *SA_AUDIT_ADMIN*

Después de activar el sistema de auditoría, pueden usarse los procedimientos *SA_AUDIT_ADMIN* para activar o desactivar la auditoría de Oracle Label Security. Para utilizar la auditoría de Oracle Label Security, debe tener asignado el rol *policy_type.*

Opciones de auditoría de Oracle Label Security

Las opciones de auditoría son las siguientes:

APPLY: Auditoría sobre la aplicación de políticas, Oracle Label Security especificadas, a tablas y esquemas.

REMOVE: Auditoría sobre la eliminación de políticas, Oracle Label Security especificadas, a tablas y esquemas.

SET: Audita la asignación de autorizaciones a usuarios, gestión de usuarios y privilegios.

PRIVILEGES: Audita el uso de todos los privilegios específicos de la política.

Activar auditoría de Oracle Label Security con SA_AUDIT_ADMIN.AUDIT

El procedimiento *AUDIT* activa la auditoría sobre política especificada.

Sintaxis:

Procedure Audit (<nombre_política> in varchar2,
<usuarios> in varchar2 default null,
<opciones> in varchar2 default null,
<tipo> in varchar2 default null,
<termina_operación> in varchar2 default null);

Donde:

<nombre_política>: Nombre de la política a auditar. La política debe existir.

<usuarios>: Campo opcional. Lista, delimitada por comas, de los usuarios que se quieren auditar. Si no se especifica, entonces todos los usuarios serán auditados.

<opciones>: Campo opcional. Lista, delimitada por comas, de las opciones que se van a auditar. Si no se especifica, entonces todas las opciones son auditadas por defecto (sin incluir la opción *PRIVILEGES*). Para auditar los privilegios, debe solicitarse explícitamente la opción *PRIVILEGES.*

<*tipo*>: Campo opcional. Su valor puede ser *BY ACCESS* (por acceso) o *BY SESSION* (por sesión). Si no se especifica, su valor por defecto es *BY SESSION*.

<*termina_operación*>: Opcional. Su valor puede ser *SUCCESSFUL* (termina la operación con éxito) o *NO SUCCESSFUL* (la operación no termina con éxito). Si no se especifica, el valor por defecto incluye ambos.

Los parámetros y opciones especificadas en el procedimiento, aplicarán a las sesiones posteriores.

Desactivar auditoría de Oracle Label Security con SA_AUDIT_ADMIN.NOAUDIT

El procedimiento *NOAUDIT* desactiva la auditoría sobre política especificada.

Sintaxis:

Procedure Noaudit (<nombre_política> in varchar2,

<usuarios> in varchar2 default null,

<opciones> in varchar2 default null);

Donde:

<*nombre_política*>: Nombre de la política a auditar. La política debe existir.

<*usuarios*>: Campo opcional. Lista, delimitada por comas, de los usuarios que se quieren dejar de auditar. Si no se especifica, entonces todos los usuarios dejarán de ser auditados.

<*opciones*>: Campo opcional. Lista, delimitada por comas, de las opciones que se van a desactivar. Si no se especifica, entonces todas las opciones por defecto son desactivadas. La desactivación de la auditoría sobre los privilegios debe ser desactivada explícitamente.

Se puede desactivar la auditoría para todas las opciones auditadas, o sólo para un subconjunto de las mismas. Todas las opciones de auditoría especificadas se desactivarán para los usuarios especificados (o todos los usuarios, si el parámetro <*usuarios*> es null).

Un caso particular a tener en cuenta, es que se realice la activación de la auditoría sobre una política Oracle Label Security para un usuario especificado, y posteriormente se lleve a cabo la activación de la auditoría sobre esta política para todos los usuarios (sin especificar el parámetro <*usuarios*>).

Sa_audit_admin.audit (‘Ejemplo_1’, ‘RRHH_MADRID’);

Sa_audit_admin.audit (‘Ejemplo_1’);

En este caso, una posterior desactivación de la auditoría sin especificar los usuarios

Sa_audit_admin.noaudit (‘Ejemplo1’);

no desactiva la auditoría que se fijó explícitamente para el usuario *‘RRHH_MADRID’.* En consecuencia, si el procedimiento *NOAUDIT* se ejecuta para todos los usuarios, la auditoría será desactivada para todos los usuarios de la política Oracle Label Security, salvo para aquellos que se estableció la auditoría de forma individual.

Los parámetros y opciones especificadas en el procedimiento, aplicarán a las sesiones posteriores.

Examinar las opciones de auditoría con la vista DBA_SA_AUDIT_OPTIONS

El usuario dispone de la vista *DBA_SA_AUDIT_OPTIONS* para poder consultar en todo momento, que usuarios están siendo auditados en cada una de las políticas de Oracle Label Security.

La vista contiene los siguientes campos:

Policy_name: nombre de la política.

User_name: nombre de usuario.

APY: columna que indica si se audita la acción de aplicar políticas a tablas y esquemas.

REM: columna que indica si se audita la acción de eliminar políticas a tablas y esquemas.

SET: columna que indica si se audita la asignación de autorizaciones de usuario, usuarios y privilegios.

PRV: columna que indica si se audita la asignación de privilegios a auditar.

El tipo de datos de los campos *APY*, *REM*, *SET* y *PRV* es una cadena de tres caracteres, cuyo valor se desglosa de la siguiente manera:

Primer carácter: La forma en que se va a auditar la ejecución correcta de una acción. Su valor puede ser por acceso (A), sesión (S) o ninguno (-).

Segundo carácter: barra separadora (/).

Tercer carácter: La forma en que se va a auditar la ejecución fallida de una acción. Su valor puede ser por acceso (A), sesión (S) o ninguno (-).

Gestión de auditoría de Etiquetas de una Política

En este punto se describen los procedimientos disponibles para la gestión de auditoría de políticas de etiqueta.

Auditoría de etiquetas de una política con SA_AUDIT_ADMIN.AUDIT_LABEL

Utilizar el procedimiento *AUDIT_LABEL* para grabar las etiquetas de la política durante la auditoría. Provoca que la etiqueta de sesión del usuario se almacene en la tabla de auditoría.

Sintaxis:

Procedure Audit_label (<nombre_política> in varchar2);

Desactivar la auditoría de etiquetas de una política con SA_AUDIT_ADMIN.NOAUDIT_LABEL

Para desactivar la auditoría de etiquetas de una política debe utilizar el procedimiento *NOAUDIT_LABEL.*

Sintaxis:

Procedure Noaudit_label (<nombre_política> in varchar2);

Estado de la auditoría de etiquetas con AUDIT_LABEL_ENABLED

Función que nos indica si se están almacenando las etiquetas de la política especificada en la tabla de auditoría.

Sintaxis:

Function Audit_label_enabled (<nombre_política> in varchar2) return Boolean;

Crear y eliminar una vista de auditoría para Oracle Label Security

Crear una vista con SA_AUDIT_ADMIN.CREATE_VIEW

Este procedimiento crea una vista sobre las pistas de auditoría de la política especificada con el nombre *DBA_<nombre_política>_AUDIT_TRAIL*. Si el nombre de la vista excede los 30 caracteres, entonces el usuario puede especificar opcionalmente un nombre más corto.

Sintaxis:

Procedure create_view (<nombre_política> in varchar2,

<nombre_vista> in varchar2 default null);

Donde:

<nombre_política>: Nombre de la política sobre la que crear la vista. La política debe existir.

<nombre_vista>: Nombre asociado a la vista, máximo 14 caracteres. En caso de que no se especifique se asignará automáticamente el nombre, según lo expuesto anteriormente.

Eliminar una vista con SA_AUDIT_ADMIN.DROP_VIEW

Procedimiento que elimina la vista sobre las pistas de auditoría de la política especificada.

Sintaxis:

Procedure drop_view (<nombre_política> in varchar2,

<nombre_vista> in varchar2 default null);

Donde:

<nombre_política>: Nombre de la política sobre la que crear la vista. La política debe existir.

<nombre_vista>: En caso de que el nombre asociado a la vista hubiera sido especificado por el usuario, entonces deberá especificarse el nombre de la misma a la hora de ser eliminada. Si no se especifica, se eliminará la vista asociada a la política.

Caso práctico

Partiendo del caso práctico del capítulo anterior, "Uso de Oracle Label Security para gestionar el etiquetado de datos". La empresa ha decidido con el fin de saber en todo momento que acciones o modificaciones se realizan sobre la política Oracle Label Security definida, habilitar las opciones de auditoría según las opciones descritas a continuación.

Del usuario de base de datos "lbacsys", se auditarán, por sesión, todas las acciones o modificaciones que se puedan hacer sobre una política, como pueden ser aplicar la política a nuevas tablas, eliminar la aplicación de la política de una tabla, modificación de autorizaciones, usuarios y privilegios.

Además, para poder ver de una forma más rápida y sencilla la pista de auditoría, la empresa ha decidido que se cree la vista "AUD_1" sobre la auditoría de la política Oracle Label Security.

Analizando el enunciado del caso práctico, se extraen los siguientes pasos a seguir:

- Activar la pista de auditoría y agregar los usuarios a los que se va a auditar, con las opciones de auditoría correspondientes. En nuestro caso, el usuario es *"lbacsys"* y las opciones de auditoría a habilitar son todas por sesión.

- Creación de la vista *"AUD_1"* sobre la pista de auditoría.

Anteriormente, en el capítulo se han mostrado los diferentes comandos a utilizar para configurar la auditoría de Oracle Label Security. Por ello, en la resolución del caso práctico, se va a realizar a través de *Oracle Enterprise Manager*.

1. Modificación de la política Oracle Label Security *"Ejemplo_1"*

En cuanto se haya accedido a Oracle Enterprise Manager con el usuario *"lbacsys"* (usuario administrador de la política). Se debe acceder a la pantalla "Políticas del Label Security". En ella, el usuario tiene que marcar la política a modificar y pulsar el botón "Editar".

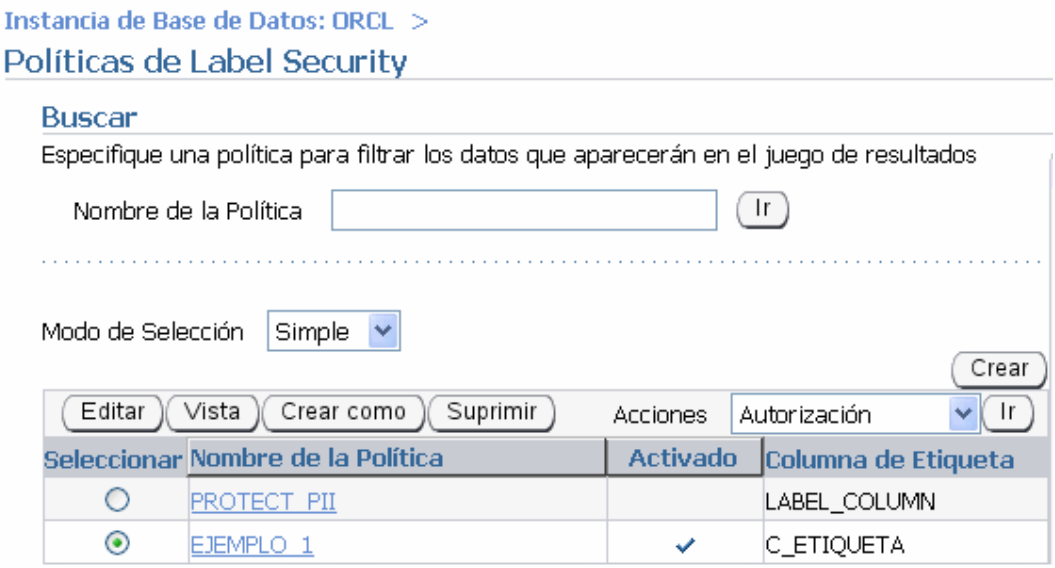

Imagen 92. Oracle Enterprise Manager. Editar política *"Ejemplo_1"*

A continuación, pulsando en la pestaña "Avanzado", con el fin de que en la pista de auditoría se incluya la etiqueta, se marcará la opción *"Incluir Etiqueta en Pista de Auditoría"*.

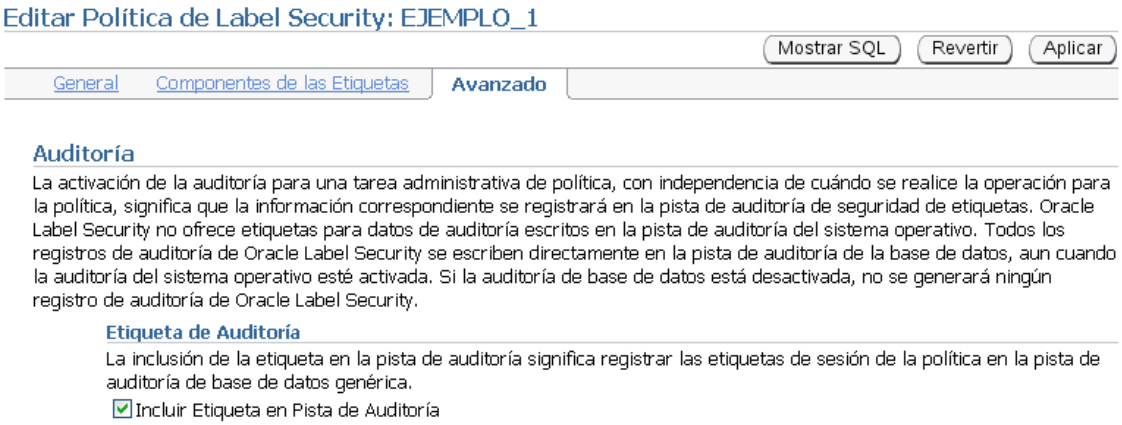

Imagen 93. Oracle Enterprise Manager. Incluir etiqueta en pista de auditoría

Finalmente, se deberá especificar los valores de auditoría para las operaciones que se quieren auditar.

En nuestro ejemplo en particular, se deben auditar todas las operaciones, por sesión del usuario *"lbacsys"*, que se pueden hacer sobre una política Oracle Label Security. Dado que el mecanismo es el mismo para configurar las opciones de auditoría de cada una de las operaciones, sólo se va a mostrar como configurar una de ellas.

En la misma pestaña que en la que se ha realizado el paso anterior, en la parte inferior de la pantalla, aparecen los valores de auditoría. Al pulsar en el desplegable "Operación" se muestran las diferentes operaciones que se pueden auditar.

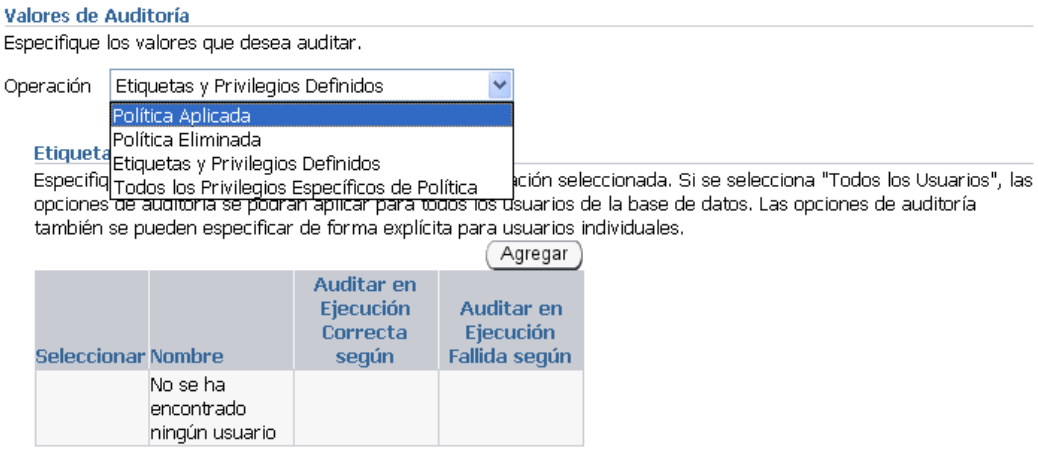

Imagen 94. Oracle Enterprise Manager. Configuración de opciones a auditar

Una vez se haya seleccionado la operación, sobre la que se quiere especificar las opciones de auditoría, habrá que pulsar el botón "Agregar", para agregar el usuario a auditar.

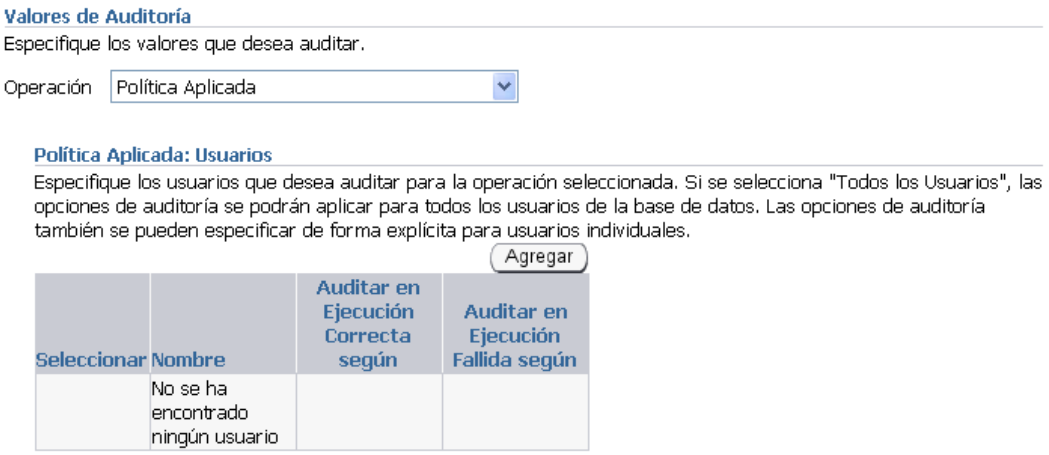

Imagen 95. Oracle Enterprise Manager. Aplicar auditoría a usuario

Al pulsar el botón "Agregar", se mostrará la siguiente pantalla en la que buscar al usuario.

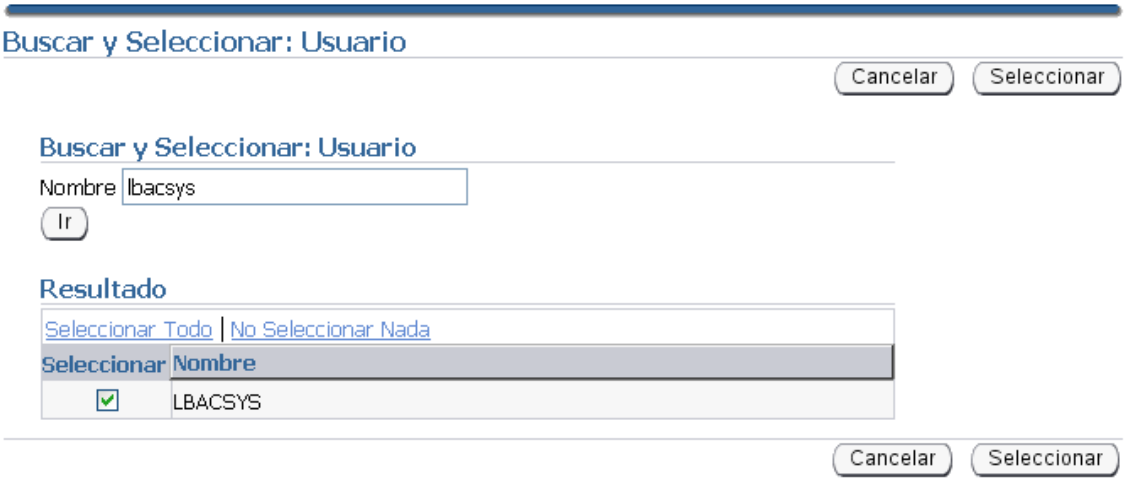

Imagen 96. Oracle Enterprise Manager. Selección de usuario a auditar

Una vez marcado el usuario, se pulsará el botón "Seleccionar". Automáticamente, se volverá a la pantalla anterior donde se podrá especificar que la auditoría se lleve a cabo por sesión o acceso.

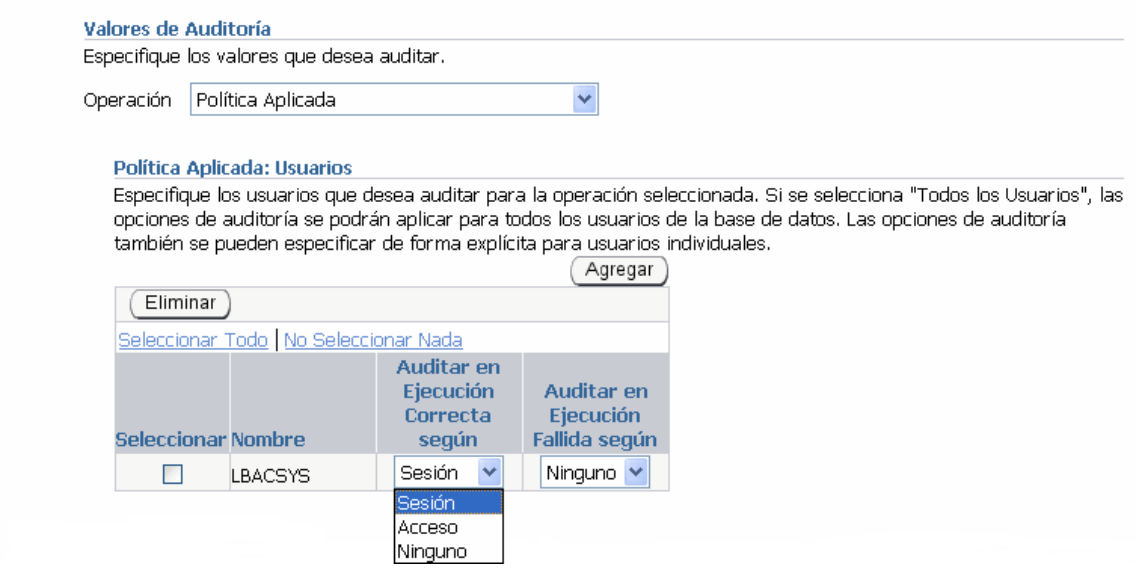

Imagen 97. Oracle Enterprise Manager. Especificación del tipo de auditoría a llevar a cabo

Como se especifica en el enunciado que se auditará por sesión, entonces se seleccionará en el desplegable por sesión, tanto en la ejecución correcta como en la ejecución fallida de la operación.

Si el usuario quiere ver el procedimiento sql que se va a ejecutar, solamente deberá pulsar el botón “Mostrar SQL”.

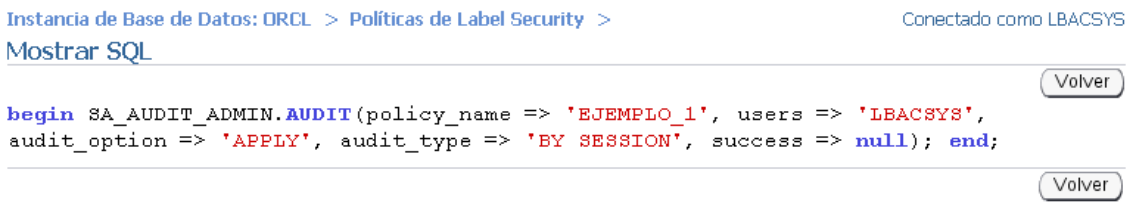

Imagen 98. Oracle Enterprise Manager. Mostrar procedimiento sql a ejecutar

Para finalizar, el usuario debe pulsar el botón “Aplicar” para que los cambios efectuados se lleven a cabo.

Se podrá comprobar que las opciones de auditoría se han configurado correctamente, seleccionando en la página principal de “Políticas de Label Security” la política deseada y pulsar a continuación el botón “Vista” para acceder a la pantalla resumen de la política.

Imagen 99. Oracle Enterprise Manager. Selección de política a visualizar

En esta pantalla, al pulsar sobre "Auditoría" se mostrará los usuarios que se están auditando por cada una de las operaciones.

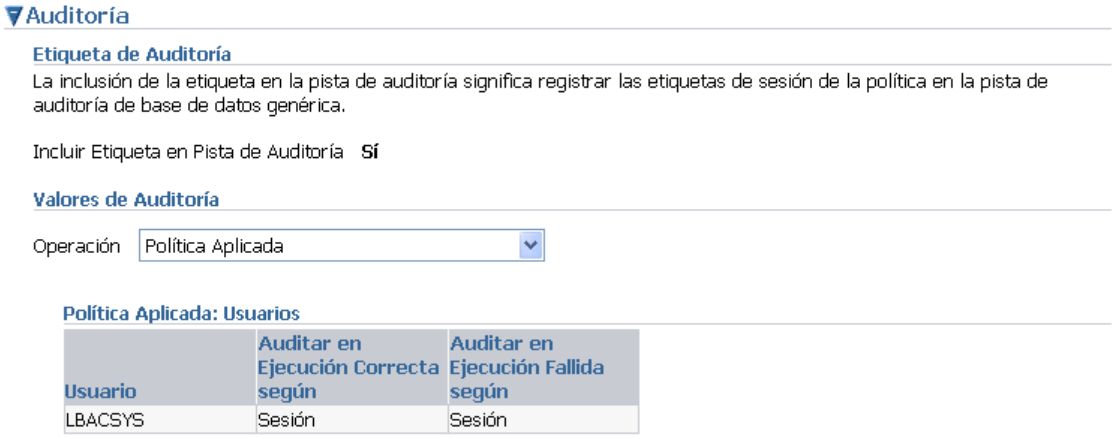

Imagen 100. Oracle Enterprise Manager. Resumen auditorías aplicadas sobre la política *"Ejemplo_1"*

2. Creación de la vista *"AUD_1"* sobre la pista de auditoría

Accediendo a SQL PLUS con un usuario que tenga permisos para utilizar el paquete de datos *SA_AUDIT_ADMIN* y administrar la política Oracle Label Security, en nuestro caso el propio usuario *"lbacsys"*. Debe lanzarse el siguiente script:

```
Begin
Sa_audit_admin.create_view('EJEMPLO_1','AUD_1');
End;
```

Al ejecutarse el script deberá salir el mensaje que se muestra en la siguiente imagen, para saber que la vista se ha creado correctamente.

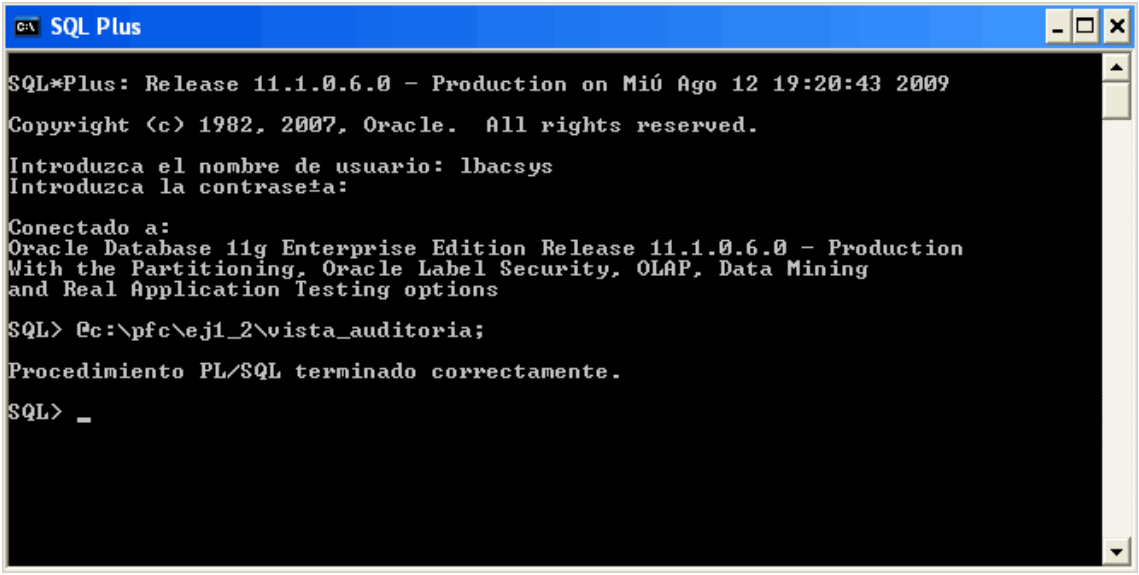

Imagen 101. Ejecución del script que crea la vista *"AUD_1"*

Pruebas

Para comprobar el buen funcionamiento de la auditoría, una vez se haya reiniciado la base de datos, la política se aplicará a la tabla *"prueba"* y posteriormente se quitará la aplicación de la política a la tabla.

Nota: La tabla *"prueba"* es una tabla con un solo campo, sin ningún valor con el fin de poder hacer las pruebas.

En la pantalla inicial de "Políticas de Label Security", se seleccionará la política *"EJEMPLO_1"*, en el desplegable "Acciones" el valor "Aplicar" y se pulsará el botón "Ir".

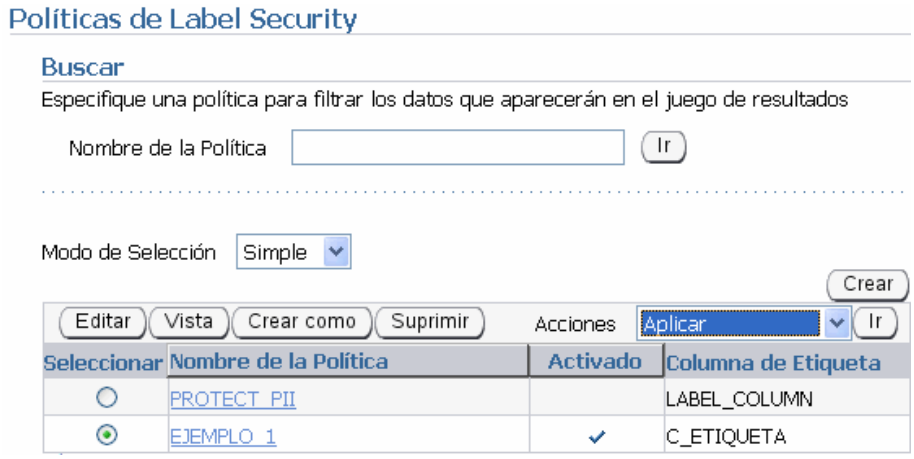

Imagen 102. Oracle Enterprise Manager. Selección política a aplicar

Se seleccionará la tabla, *"prueba"* en nuestro ejemplo, sobre la que se quiere aplicar la política y se pulsará el botón "Aceptar".

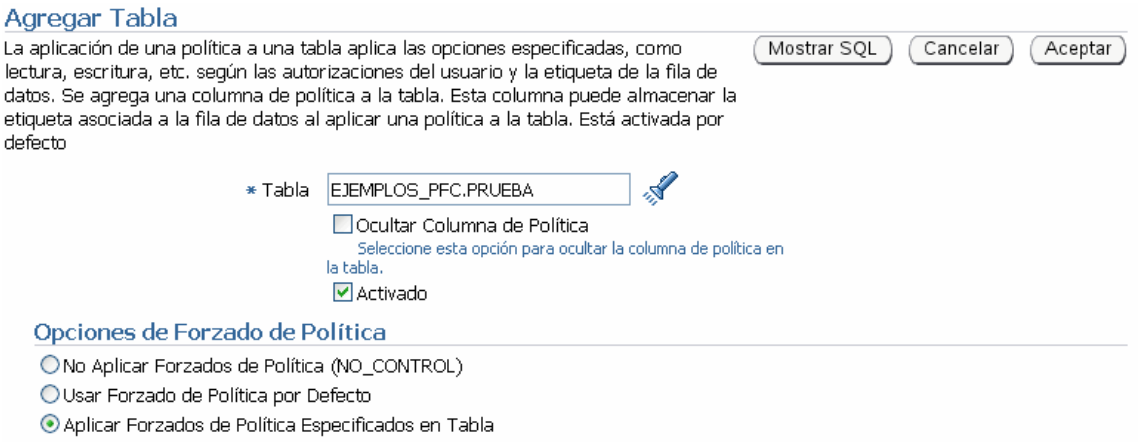

Imagen 103. Oracle Enterprise Manager. Aplicar política *"Ejemplo_1"* a la tabla *"prueba"*

Posteriormente, se eliminará la aplicación de la política a la tabla *"prueba"* y se ejecutará la siguiente consulta sobre la vista *"AUD_1"*.

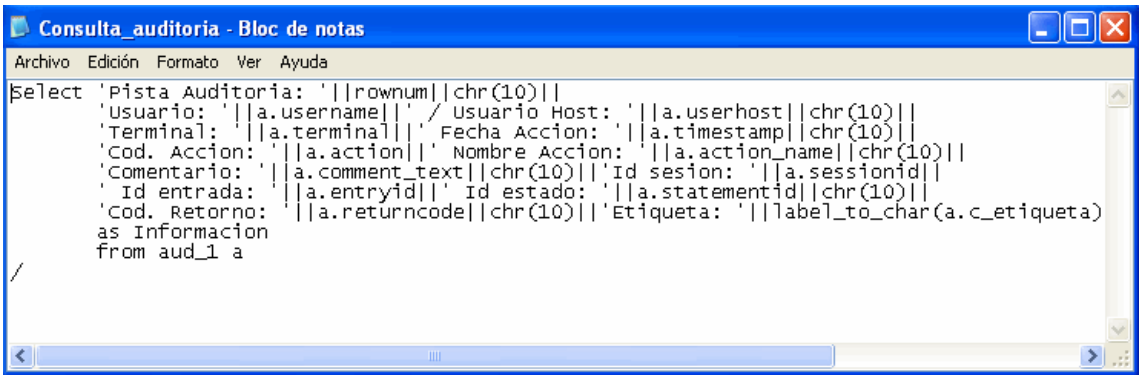

Imagen 104. Consulta sobre la vista *"AUD_1"*

Como resultado se obtiene:

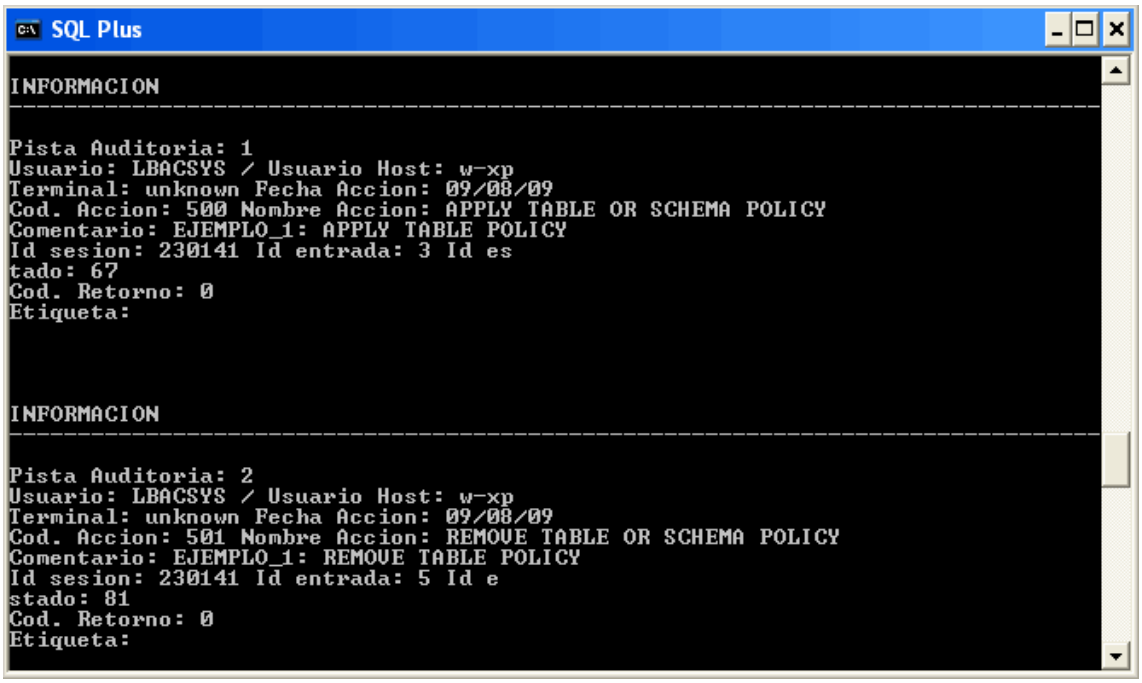

Imagen 105. Resultado consulta sobre la vista *"AUD_1"* (Ejemplo 1)

Donde se puede ver como se ha aplicado la política a una tabla y posteriormente se ha eliminado la aplicación de la política a una tabla.

Con el fin de realizar otra prueba, se va a modificar la etiqueta del usuario *"INFO_ASTURIAS"*.

En la pantalla inicial de "Políticas de Label Security", se seleccionará la política *"EJEMPLO_1"*, en el desplegable "Acciones" el valor "Autorización" y se pulsará el botón "Ir".

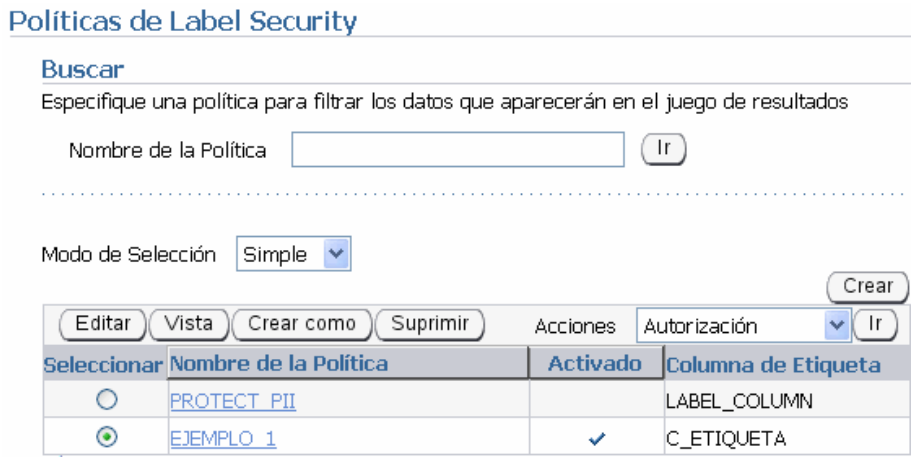

Imagen 106. Oracle Enterprise Manager. Editar autorizaciones de la política *"Ejemplo_1"*

Se seleccionará el usuario *"INFO_ASTURIAS"* y se pulsará el botón "Editar".

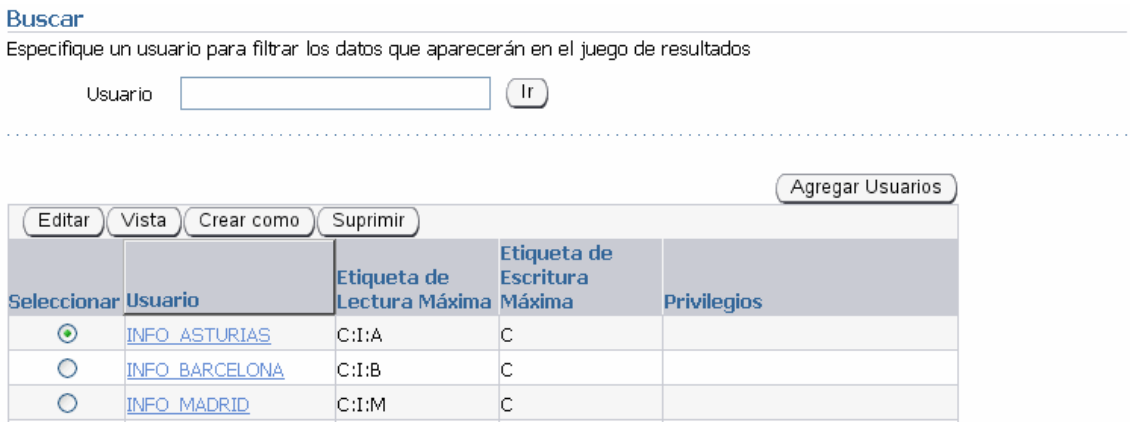

Imagen 107. Oracle Enterprise Manager. Edición de la autorización del usuario *"INFO_ASTURIAS"*

En la pestaña "Componentes de Etiqueta" en la sección de "Grupos" se seleccionará el grupo *"A"* y se pulsará el botón "Eliminar".

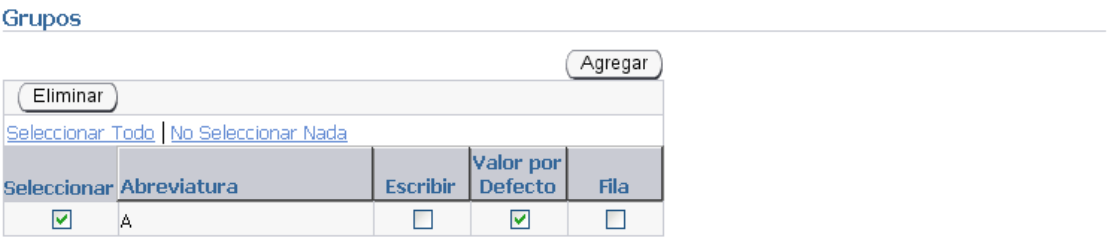

Imagen 108. Oracle Enterprise Manager. Eliminar grupo *"A"* de la autorización del usuario *"INFO_ASTURIAS"*

Para finalizar la operación, se deberá pulsar el botón "Aplicar".

Ejecutando la consulta de la prueba anterior, se podrá ver como el usuario *"lbacsys"* ha realizado una modificación en las autorizaciones/privilegios de una etiqueta.

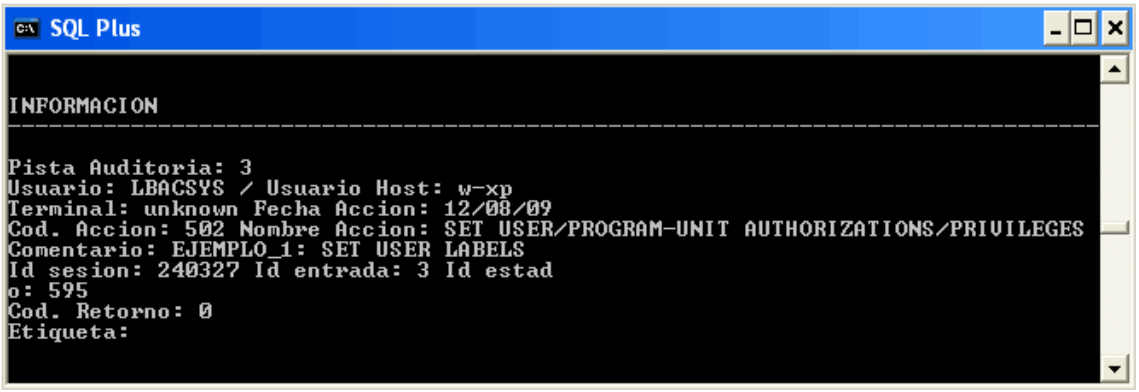

Imagen 109. Resultado consulta sobre la vista *"AUD_1"* (Ejemplo 2)

Nota: El campo "Etiqueta" aparece vacío, porque para el usuario *"lbacsys"* no tiene etiqueta de sesión dentro de la política.

Conclusiones

El objetivo de este Proyecto Fin de Carrera se divide en dos. El primer objetivo era mostrar al lector en qué factores se divide la obtención de un producto de calidad en una base de datos.

Para la realización de este primer objetivo, antes de adentrarse en qué factores se divide la obtención de la calidad en una base de datos, ha sido necesario estudiar los diferentes aspectos relacionados con el concepto del término calidad en cuanto al software:

- Qué significado tiene el término calidad en el ámbito del software, sus características principales, los tipos de calidad existentes, junto con los costes que conlleva implantar un sistema de gestión de calidad.

- Cuáles son los principios de la gestión de la calidad.

- Con el fin de asegurar la calidad, cuáles son los puntos más importantes de las normas ISO 9001 e ISO 90003 que una organización debe llevar a cabo para asegurar la obtención de un producto de calidad.

Una vez analizados estos puntos, el lector conocerá las nociones básicas que toda organización debe llevar a cabo para poder conseguir y asegurar a sus clientes que sus productos son de calidad.

A continuación, se ha realizado la introducción a Oracle, teniendo en cuenta que en una base de datos proporcionar seguridad a los datos almacenados está muy relacionado con la calidad, se analizan los diferentes puntos de control o factores principales a tener en cuenta, como puede ser la autenticación y autorización, seguridad tras la instalación… Hasta adentrar al lector en las diferentes características que nos proporciona el SGBD Oracle. Dentro de estas características hay que resaltar la seguridad a nivel de fila que proporciona Oracle, ya que el segundo y principal objetivo del proyecto es realizar un estudio de este tema en cuestión.

Adentrándonos en el segundo objetivo del proyecto, conseguir seguridad a nivel de fila es altamente complicado mediante la implementación de programas, pero Oracle nos ofrece la herramienta Oracle Label Security, de forma que el usuario pueda definir políticas que proporcionen una seguridad a nivel de filas mediante el uso de etiquetas de sensibilidad.

Se ha realizado un estudio pormenorizado sobre la utilización de la herramienta Oracle Label Security, partiendo de las nociones básicas hasta analizar de una forma más detallada las diferentes opciones que nos proporciona esta herramienta. Además de la información teórica, se han implementado diversos ejemplos con el fin de proporcionar una visión práctica de los diferentes temas tratados.

En este punto, tengo que decir que en un principio intenté realizar la instalación de esta herramienta con la versión 10g de Oracle. Para ello era necesario crear un sistema de cluster que se comunicaran entre sí, pero una vez creadas las máquinas virtuales necesarias, cuando llegaba al último paso del proceso de instalación se producía un error debido a que ambas máquinas virtuales compartían el mismo disco duro. Finalmente esta opción tuve que abandonarla, porque no encontraba solución al problema, y tuve que realizar la instalación de esta herramienta con la versión 11g de Oracle.

Considero que se han cumplido los objetivos del proyecto abordándolos de una manera ordenada y objetiva. En el primer objetivo partiendo de la definición de calidad en el software hasta llegar a los diferentes factores que deben tenerse en cuenta para tener una base de datos de calidad. Para la consecución del segundo objetivo, se ha partido de lo más básico hasta llegar a lo más complejo, intentado abordar las diferentes opciones funcionales de una manera ordenada, mostrando primeramente los conocimientos teóricos necesarios para su comprensión y posteriormente la implementación de casos prácticos a modo de guía, con imágenes de los pasos a seguir para facilitar el entendimiento a la hora de realizar la implementación, gráficos, así como anexos complementarios.

Al iniciar el Proyecto Fin de Carrera, en un principio quería realizar un estudio acerca de la calidad del desarrollo de una aplicación software en una base de datos, pero una vez adentrado en el estudio de la calidad y seguridad en bases de datos, me fascinó la manera en que se podía conseguir la seguridad a nivel de filas en una bases de datos y como el SGBD Oracle dispone de una herramienta muy útil para este fin. Así que decidí encaminar el proyecto hacía este tema. Para ello he tenido que adquirir nuevos conocimientos y a pesar de no ser complejos, a veces fue laborioso encontrar la información precisa.

Tras la realización de este proyecto, he adquirido un conocimiento básico de cómo una organización debe llevar a cabo diversas tareas para asegurar la calidad.

En cuanto al objetivo principal del proyecto, he adquirido conocimientos acerca del manejo y comprensión de una herramienta como Oracle Label Security, la facilidad que proporciona al usuario para implementar una política de seguridad, cómo administrarla, la gestión de autorización de usuarios, la gestión de etiquetas, las diferentes opciones de aplicación, etc.

Líneas de investigación futuras

En un sistema gestor de bases de datos, el objetivo primordial es mantener la seguridad de los datos que se manipulan y tenerlos disponibles en cualquier momento.

Desde que se fundó a finales de los años 70, Oracle siempre ha intentado estar a la vanguardia con el fin de proporcionar innovaciones en cuanto a la seguridad y mejorar el funcionamiento de su sistema gestor de base de datos.

Como se ha expuesto a lo largo del proyecto el SGBD Oracle proporciona la herramienta Oracle Label Security, con el fin de que el usuario o administrador de la base de datos proporcione la seguridad a nivel de fila necesaria. Pero no solamente proporciona esta herramienta, Oracle también proporciona otras herramientas que sería interesante someterlas en un futuro a estudio, como:

- Oracle Database Vault

  Herramienta que proporciona características de seguridad avanzadas para restringir el acceso a la base de datos.

- Oracle Advanced Security

  Herramienta que ofrece protección de datos a fin de asegurar tanto la información inactiva como la que se encuentra en tránsito. Combina el cifrado de red, cifrado transparente de datos de todo tipo y un medio de autenticación sólido.

- Oracle Audit Vault

  Herramienta que proporciona un sistema automatizado que recopila y analiza los datos de auditoría de múltiples sistemas y captura la información clave de los registros de transacciones.

Otros factores a analizar pueden ser como interactúa Oracle con otros sistemas para mantener la seguridad, o comparar el SGBD Oracle con otros SGBD, como puede ser Microsoft SQL Server.

Anexo I: Gestión de autorizaciones de usuario

Introducción

Anexo en el que se exponen, en un primer apartado, las características de las autorizaciones de etiquetas de usuarios de Oracle Label Security. En un segundo apartado, se muestran los diferentes tipos de autorizaciones especiales que se pueden otorgar a un usuario.

Características de autorización de etiquetas de usuario

Las autorizaciones de usuarios Oracle Label Security deben ser establecidas por un administrador de seguridad antes de que un usuario de aplicaciones pueda acceder a una tabla de aplicaciones protegida por Oracle Label Security. Las autorizaciones de etiquetas de los usuarios de Oracle Label Security se definen de la siguiente manera:

Nivel Máximo – El nivel máximo de sensibilidad al que un usuario puede acceder. En un entorno *hosting,* solo puede haber un nivel único. En ámbitos de gobierno y defensa, pueden definirse cuatro o cinco niveles.

Nivel Mínimo – El nivel mínimo de sensibilidad en el que un usuario puede escribir los datos. Por ejemplo, un administrador puede evitar que los usuarios etiqueten los datos como *Públicos* al asignar un nivel mínimo de *Company Confidencial (Confidencial para la Empresa).*

Nivel por Defecto – El nivel utilizado por defecto cuando un usuario se conecta a la base de datos. Por ejemplo, un usuario puede configurar su nivel por defecto en *Secret*. Cuando se conecta al sistema, el nivel por defecto se iniciará en *Secret*.

Nivel de Filas – El nivel utilizado para etiquetar los datos ingresados por el usuario en la base de datos a través de la aplicación o directamente mediante una herramienta como SQL*Plus.

Compartimentos de Lectura – El grupo de compartimentos asignados al usuario y utilizados durante la mediación de acceso READ. Por ejemplo, si un usuario tiene compartimentos *A, B y C*, podría ver los datos con compartimentos *A y B* pero no los datos con compartimentos *A, B, C y D.*

Compartimentos de Escritura – El grupo de compartimentos asignados al usuario y utilizados durante la mediación de acceso WRITE. Por ejemplo, se podría otorgar a un usuario acceso READ y WRITE a los compartimentos *A y B* pero acceso READ-ONLY (de solo lectura) al compartimento *C.* Si el registro de una aplicación tuviese etiquetas en los compartimentos *A, B y C,* el usuario no estaría autorizado a actualizar el registro ya que no tiene el acceso WRITE (de escritura) para el compartimento *C*.

Grupos de Lectura – El conjunto de grupos asignado a un usuario y utilizado durante la mediación de acceso READ. Por ejemplo, si un usuario tuviese el grupo *Manager (administrador)*, podría ver los datos del grupo *Manager* pero no los datos del grupo *Senior VP (VP Sénior).*

Grupos de Escritura – El conjunto de grupos asignado al usuario y utilizado durante la mediación de acceso WRITE. Por ejemplo, un usuario podría estar autorizado con el acceso READ y WRITE al grupo *Senior VP* pero con el acceso READ-ONLY al grupo *Manager*. Si el registro de una aplicación tuviese la etiqueta de un grupo único, *Manager*, el usuario no estaría autorizado a actualizar el registro ya que no tendría acceso WRITE al grupo *Manager*.

Autorizaciones especiales para usuarios

READ – La autorización READ permite que un usuario acceda a todos los datos protegidos por Oracle Label Security, sin embargo, la mediación de acceso todavía se aplica en las operaciones UPDATE, INSERT y DELETE. Oracle Label Security no realiza la verificación de mediaciones en las operaciones SELECT.

FULL – La autorización FULL desactiva todo tipo de mediación de acceso de Oracle Label Security. Un usuario con la autorización FULL puede realizar operaciones SELECT, UPATE, INSERT y DELETE sin autorización de etiquetas. Ha de tenerse en cuenta que las autorizaciones Oracle SYSTEM y OBJECT todavía se aplican. Por ejemplo, un usuario todavía debe tener SELECT en la tabla de aplicaciones. La autorización FULL desactiva la verificación de mediación de acceso en el nivel de fila individual.

WRITEDOWN – La autorización WRITEDOWN permite que un usuario modifique el componente de nivel de una etiqueta y reduce la sensibilidad de la etiqueta. Por ejemplo, los datos de aplicaciones con etiquetas *Top Secret: Alpha, Beta* podrían cambiar a *Secret: Alpha, Beta*. Esta autorización solo es aplicable a políticas que utilizan la opción de actualización de etiquetas.

WRITEUP – La autorización WRITEUP permite que un usuario modifique el componente de nivel de una etiqueta y aumente la sensibilidad de la etiqueta. Por ejemplo, los datos de aplicaciones con etiquetas *Secret: Alpha, Beta* podrían cambiar a *Top Secret: Alpha, Beta*. Ha de tenerse en cuenta que la autorización de la etiqueta *Nivel Máximo* asignada al usuario limitará la modificación. Esta autorización solo es aplicable a políticas que utilizan la opción de actualización de etiquetas.

WRITEACROSS – La autorización WRITEACROSS permite que un usuario modifique los compartimentos y grupos de una etiqueta en cualquier compartimento o grupo válido de Oracle Label Security para la política. Por ejemplo, los datos de aplicaciones con la etiqueta *Secret: Alpha, Beta* pueden ser modificados a *Secret: Alpha, Beta, Delta* incluso si el usuario no está autorizado para el compartimento *Delta*. Esta autorización solo es aplicable a políticas que utilizan la opción de actualización de etiquetas.

PROFILE ACCESS – La autorización PROFILE ACCESS permite que un usuario asuma las autorizaciones Oracle Label Security de cualquier otro usuario. Por ejemplo, el usuario *Scott* que tiene acceso a los compartimentos *A,B y C* podría asumir el perfil de usuario *Joe* que tiene acceso a los compartimentos *A,B, C y D.* Esta funcionalidad podría ser útil en un ámbito donde una aplicación utiliza una sola cuenta de aplicaciones para todos los usuarios de aplicaciones. La cuenta de aplicaciones podría utilizar la autorización PROFILE ACCESS para asumir inmediatamente un perfil designado para seguridad de etiquetas cuando un usuario de aplicaciones se conecta al sistema.

Anexo II: Administración de etiquetas de usuario y privilegios

Introducción

En este punto, se va a exponer como utilizar los paquetes de Oracle Label Security para la administración de etiquetas de usuario y gestionar los privilegios.

Para poder gestionar las etiquetas de usuario y los privilegios, se debe tener el permiso *execute* sobre el paquete *SA_USER_ADMIN*, y haber sido concedido el rol *policy_DBA*.

El paquete *SA_USER_ADMIN* proporciona las funciones para la gestión de Oracle Label Security mediante el uso de atributos de seguridad. Contiene una serie de procedimientos para la gestión de etiquetas de usuario por componente, es decir, especificando niveles de usuario, los compartimentos y los grupos.

Toda la información de las etiquetas y privilegios se almacena en las tablas de diccionario de datos de Oracle Label Security. Cuando un usuario se conecta a la base de datos, sus etiquetas de sesión se establecen sobre la información almacenada en el diccionario de datos de Oracle Label Security.

Gestión de etiquetas de usuario por componente, con *SA_USER_ADMIN*

Los siguientes procedimientos, del paquete *SA_USER_ADMIN*, permiten la gestión de componentes de etiquetas de usuario.

SA_USER_ADMIN.SET_LEVES

Procedimiento que permite asignar el nivel mínimo y máximo a un usuario e identifica los valores por defecto para la etiqueta de sesión y la etiqueta de fila de usuario.

Sintaxis:

Procedure set_levels (<nombre_política> in varchar2,
<nombre_usuario> in varchar2,
<nivel_máximo> in varchar2,
<nivel_mínimo> in varchar2 default null,
<nivel_por_defecto> in varchar2 default null,
<nivel_fila> in varchar2 default null);

Donde:

<*nivel_máximo*> es el nivel más alto de acceso de lectura y escritura.

<*nivel_mínimo*> es el nivel más bajo de escritura. Si <*nivel_mínimo*> es nulo, entonces se establece el nivel más bajo definido por la política.

Si <*nivel_por_defecto*> no se especifica, entonces se establece el <*nivel_máximo*>.

Si <*nivel_fila*> no se especifica, entonces se establece el <*nivel_por_defecto*>.

SA_USER_ADMIN.SET_COMPARTMENTS

Procedimiento que asigna compartimentos a un usuario y determina los valores por defecto para la etiqueta de sesión y la etiqueta de fila del usuario.

Todos los usuarios deben tener sus niveles asignados antes de que la autorización de compartimentos pueda ser establecida.

En caso de que se especifique compartimentos de escritura, estos deben ser un subconjunto de los compartimentos de lectura.

Sintaxis:

Procedure set_compartments(<nombre_política> in varcahr2,
<nombre_usuario> in varchar2,
<copartimentos_lectura> in varchar2,
<compartimentos_escritura> in varchar2 default null,
<compartimentos_defecto> in varchar2 default null,
<fila_compartimentos> in varcahr2 default null);

Donde:

<*compartimentos_lectura*> es una lista, delimitada por comas, de compartimentos autorizados para el acceso de lectura.

<*compartimentos_escritura*> es una lista, delimitada por comas, de los compartimentos autorizados para el acceso de escritura. Este debe ser un subconjunto de <*compartimentos_lectura*>. Si <*compartimentos_escritura*> es nulo, entonces se establecen los <*compartimentos_lectura*>.

<*compartimentos_defecto*> especifica el valor por defecto de los compartimentos. Este debe ser un subconjunto de <*compartimentos_lectura*>. Si <*compartimentos_defecto*> es nulo, entonces se establecen los <*compartimentos_lectura*>.

<*fila_compartimentos*> especifica la fila de compartimentos. Este debe ser un subconjunto de los <*compartimentos_escritura*> y <*compartimentos_defecto*>. Si <*fila_compartimentos*> es nulo, entonces se fija los componentes de <*compartimentos_defecto*> que están autorizados para el acceso de escritura.

SA_USER_ADMIN.SET_GROUPS

Procedimiento que asigna grupos a un usuario y determina los valores por defecto para la etiqueta de sesión y la etiqueta de fila del usuario.

Todos los usuarios deben tener sus niveles asignados antes de que la autorización de grupos pueda ser establecida.

Sintaxis:

Procedure set_groups(<nombre_política> in varchar2,
<nombre_usuario> in varchar2,
<grupos_lectura> in varchar2,
<grupos_escritura> in varchar2 default null,
<grupos_defecto> in varchar2 default null,
<fila_grupos> in varcahr2 default null);

Donde:

<*grupos_lectura*> es una lista, delimitada por comas, de grupos autorizados para el acceso de lectura.

<*grupo_escritura*> es una lista, delimitada por comas, de los grupos autorizados para el acceso de escritura. Este debe ser un subconjunto de <*grupos_lectura*>. Si <*grupos_escritura*> es nulo, entonces se establecen los <*grupos_lectura*>, al igual que en el procedimiento *set_components*.

<*grupos_defecto*> especifica el valor por defecto de los grupos. Este debe ser un subconjunto de <*grupos_lectura*>. Si <*grupos_defecto*> es nulo, entonces se establecen los <*grupo_lectura*>.

<*fila_grupos*> especifica la fila de grupos. Este debe ser un subconjunto de los <*grupos_escritura*> y <*grupos_defecto*>. Si <*fila_grupos*> es nulo, entonces se fija los grupos de <*grupos_defecto*> que están autorizados para el acceso de escritura.

SA_USER_ADMIN.ALTER_COMPARTMENTS y ADD_COMPARTMENTS

Alter_compartments

Procedimiento encargado de cambiar el acceso de escritura, el indicador de etiqueta por defecto y el indicador de etiqueta de fila para cada uno de los compartimentos de la lista.

Sintaxis:

Procedure alter_compartments (<nombre_política> in varchar2,
<nombre_usuario> in varchar2,
<compartimentos> in varchar2,
<modo_acceso> in varchar2 default null,
<indicador_et_defecto> in varchar2 default null,
<indicador_fil_defecto> in varchar2 default null);

Add_compartments

Este procedimiento se encarga de agregar compartimentos a las autorizaciones de usuarios, indicando si compartimentos están autorizados para escribir y leer.

Sintaxis:

Procedure add_compartments (<nombre_política> in varchar2,
<nombre_usuario> in varchar2,
<compartimentos> in varchar2,
<modo_acceso> in varchar2 default null,
<indicador_et_defecto> in varchar2 default null,
<indicador_fil_defecto> in varchar2 default null);

Donde:

<*compartimentos*> es una lista, delimitada por comas, de los compartimentos a modificar o agregar.

<*modo_acceso*> sirve para especificar el tipo de acceso que se autoriza al usuario, su valor puede ser:

SA_UTL.READ_ONLY para indicar que no tiene acceso de escritura.

SA_UTL.READ_WRITE para indicar que tiene acceso de escritura.

Si no se especifica un valor, para el procedimiento *alter_compartments* el modo de acceso para el compartimento será inalterable. Mientras, que para el procedimiento *add_compartments*, se establecerá por defecto el valor SA_UTL.READ_ONLY.

<*indicador_et_defecto*> se utiliza para especificar si los compartimentos deben tener un valor por defecto (Y/N).

Si no se especifica un valor, para el procedimiento *alter_compartments* entonces <*indicador_et_defecto*> para el compartimento es inalterable. Sin embargo, para el procedimiento *add_compartments*, se establecerá "Y" por defecto.

<*indicador_fil_defecto*> se utiliza para especificar si los compartimentos deben estar en la etiqueta de la fila (Y/N).

Si no se especifica un valor, para el procedimiento *alter_compartments* entonces <*indicador_fil_defecto*> para el compartimento es inalterable. Mientras, que para el procedimiento *add_compartments*, se establecerá "N" por defecto.

SA_USER_ADMIN.DROP_COMPARTMENTS y DROP_ALL_COMPARTMENTS

Drop_compartments

Procedimiento encargado de eliminar los compartimentos especificados dentro de los compartimentos autorizados de un usuario.

Sintaxis:

Procedure drop_compartments (<nombre_política> in varchar2,
<nombre_usuario> in varchar2,
<compartimentos> in varchar2);

Donde:

<compartimentos> es una lista, delimitada por comas, de los compartimentos a eliminar.

Drop_all_compartments

Procedimiento encargado de eliminar todos los compartimentos que tiene autorizado un usuario.

Sintaxis:

Procedure drop_all_compartments (<nombre_política> in varchar2,
<nombre_usuario> in varchar2);

SA_USER_ADMIN.ADD_GROUPS y ALTER_GROUPS

Add_groups

Procedimiento encargado de añadir grupos a un usuario, indicando si se autoriza al usuario para lectura y escritura.

Sintaxis:

Procedure add_groups (<nombre_política> in varchar2,
<nombre_usuario> in varchar2,
<grupos> in varchar2,
<modo_acceso> in varchar2 default null,
<indicador_et_defecto> in varchar2 default null,
<indicador_fil_defecto> in varchar2 default null);

Alter_groups

Procedimiento encargado de cambiar el modo de acceso, el indicador de etiqueta por defecto, y el indicador de la etiqueta de fila para cada uno de los grupos especificados.

Sintaxis:

Procedure alter_groups (<nombre_política> in varchar2,
<nombre_usuario> in varchar2,
<grupos> in varchar2,
<modo_acceso> in varchar2 default null,
<indicador_et_defecto> in varchar2 default null,
<indicador_fil_defecto> in varchar2 default null);

Donde:

<*grupos*> es una lista, delimitada por comas, de los grupos a modificar o agregar.

<*modo_acceso*> sirve para especificar el tipo de acceso que se autoriza al usuario, su valor puede ser:

SA_UTL.READ_ONLY para indicar que no tiene acceso de escritura.

SA_UTL.READ_WRITE para indicar que tiene acceso de escritura.

Si no se especifica un valor, para el procedimiento *add_groups* el modo de acceso para el grupo será SA_UTL.READ_ONLY. Mientras, que para el procedimiento *alter_groups*, el <*modo_acceso*> será inalterable.

<*indicador_et_defecto*> se utiliza para especificar si los grupos deben tener un valor por defecto (Y/N).

Si no se especifica un valor, para el procedimiento *add_groups*, se establecerá "Y" por defecto. Sin embargo, para el procedimiento *alter_groups* el <*indicador_et_defecto*> para el grupo es inalterable.

<*indicador_fil_defecto*> se utiliza para especificar si los grupos deben estar en la etiqueta de la fila (Y/N).

Si no se especifica un valor, para el procedimiento *add_groups*, se establecerá "N" por defecto. Mientras, que para el procedimiento *alter_groups* entonces <*indicador_fil_defecto*> para el grupo es inalterable.

SA_USER_ADMIN.DROP_GROUPS y DROP_ALL_GROUPS

Drop_groups

Procedimiento encargado de eliminar los grupos especificados dentro de los grupos autorizados de un usuario.

Sintaxis:

Procedure drop_groups (<nombre_política> in varchar2,
<nombre_usuario> in varchar2,
<grupos> in varchar2);

Donde:

<grupos> es una lista, delimitada por comas, de los grupos a eliminar.

Drop_all_groups

Procedimiento encargado de eliminar todos los grupos que tiene autorizado un usuario.

Sintaxis:

Procedure drop_all_groups (<nombre_política> in varchar2,
<nombre_usuario> in varchar2);

Gestión de etiquetas de usuario por la cadena de caracteres que representa a la etiqueta, con *SA_USER_ADMIN*

Los siguientes procedimientos permiten gestionar las etiquetas de usuario, especificando la cadena de caracteres que representa a la etiqueta.

SA_USER_ADMIN.SET_USER_LABELS

Procedimiento que establece los niveles de usuario, compartimentos y grupos usados en un conjunto de etiquetas, en lugar de los componentes individuales.

Sintaxis:

Procedure set_user_labels (<nombre_política> in varchar2,
<nombre_usuario> in varchar2,
<max_nivel_lectura> in varchar2,
<max_nivel_escritura> in varchar2 default null,
<min_nivel_escritura> in varchar2 default null,
<etiqueta> in varchar2 default null,
<etiqueta_fila> in varchar2 default null);

Donde:

<*max_nivel_lectura*> especifica la cadena de caracteres que representa a la etiqueta que es usada como la etiqueta máxima de lectura autorizada a un usuario. La etiqueta se compone del nivel máximo asociado al usuario, compartimentos y grupos autorizados para el acceso de lectura.

<*max_nivel_escritura*> especifica la cadena de caracteres que representa a la etiqueta que es usada como la etiqueta máxima de escritura autorizada a un usuario. La etiqueta se compone del nivel máximo asociado al usuario, compartimentos y grupos autorizados para el acceso de escritura. Si <*max_nivel_escritura*> no se especifica, entonces se establece como <*max_nivel_escritura*> el <*max_nivel_lectura*>.

<*min_nivel_escritura*> especifica la cadena de caracteres que representa a la etiqueta que es usada como la etiqueta mínima de escritura autorizada a un usuario. Solamente se especifica el nivel, sin compartimentos o grupos. Si no se especifica, entonces se establece el nivel más bajo definido por la política, sin compartimentos o grupos.

<*etiqueta*> especifica la cadena de caracteres que representa a la etiqueta que se utiliza para inicializar la etiqueta de la sesión de usuario, incluyendo el nivel, los compartimentos y grupos (un subconjunto de <*max_nivel_lectura*>). Si no se especifica, entonces se establece como <*etiqueta*> el <*max_nivel_lectura*>.

<*etiqueta_fila*> especifica la cadena de caracteres que representa la etiqueta que se utiliza para inicializar la etiqueta de fila del programa. Incluye nivel, compartimentos y grupos (subconjunto de <*max_nivel_escritura*> y <*etiqueta*>). Si no se especifica, entonces se establece como <*etiqueta_fila*> los compartimentos y grupos autorizados para el acceso de escritura.

SA_USER_ADMIN.SET_DEFAULT_LABEL

Procedimiento que establece al periodo de sesión inicial de usuario la etiqueta especificada.

Sintaxis:

Procedure set_default_label (<nombre_política> in varchar2,
<nombre_usuario> in varchar2,
<etiqueta> in varchar2);

Donde:

<etiqueta> especifica la cadena de caracteres que representa la etiqueta que se utiliza para inicializar la etiqueta de usuario por defecto. La etiqueta puede contener cualquier compartimento y grupos que están autorizados para el acceso de lectura.

A la hora de definir la etiqueta hay que tener en cuenta:

- Cualquier nivel debe ser igual o inferior al de la etiqueta máxima, e igual o superior al de la etiqueta mínima.

- Incluir todos los compartimentos en la lista de compartimentos autorizados.

- Incluir todos los grupos en la lista de grupos autorizados.

La etiqueta de fila debe estar dominada por la nueva etiqueta de escritura, que se obtiene como resultado al reiniciar el período de sesiones de etiqueta. Si esta condición no se cumple, entonces el procedimiento *set_default_label* fallará.

SA_USER_ADMIN.SET_ROW_LABEL

Procedimiento para configurar la etiqueta de fila de usuario.

Sintaxis:

Procedure set_row_label (<nombre_política> in varchar2,
<nombre_usuario> in varchar2,
<etiqueta_fila> in varchar2);

Donde:

<etiqueta_fila> especifica la cadena de caracteres que representa la etiqueta que se utiliza para inicializar la etiqueta de fila del usuario. La etiqueta debe contener únicamente los compartimentos y los grupos de la etiqueta por defecto que está autorizada para el acceso de escritura.

El usuario puede establecer una etiqueta de fila independiente, pero sólo si:

- El nivel es inferior o igual al nivel del período de sesiones de etiqueta, y mayor o igual al nivel mínimo de usuario.

- Incluye un subconjunto de los compartimentos y los grupos de la sesión de etiqueta, para la cual el usuario tiene autorizado el acceso de escritura.

SA_USER_ADMIN.DROP_USER_ACCES

Procedimiento que elimina todas las autorizaciones y privilegios de Oracle Label Security para el usuario especificado.

Sintaxis:

Procedure drop_user_acces (<nombre_política> in varchar2,
<nombre_usuario> in varchar2);

Gestión de privilegios de usuario con *SA_USER_ADMIN.SET_USER_PRIVS*

Procedimiento que establece los privilegios específicos para un usuario dentro de una política. Estos privilegios no se harán efectivos en el período actual de sesión. Sin embargo, entrarán en vigor la próxima vez que el usuario inicia sesión. El nuevo conjunto de privilegios reemplaza cualquier privilegio existente hasta ese momento. El valor *null* para el parámetro <*privilegios*> elimina todos los privilegios del usuario para la política especificada.

Para poder asignar a los usuarios privilegios dentro de una política, debe tener el privilegio de *ejecución* del paquete *SA_USER_ADMIN*, y debe haberse concedido el rol de *policy_DBA* (administrador de la política).

Sintaxis:

Procedure set_user_privs (<nombre_política> in varchar2,
<nomber_usuario> in varchar2,
<privilegios> in varchar2);

Donde:

<*privilegios*> es una cadena, separada por comas, de privilegios específicos dentro de la política.

Configuración de etiquetas y privilegios con *SA_SESSION.SET_ACCES_PROFILE*

Procedimiento que establece las autorizaciones y privilegios Oracle Label Security en la sesión de base de datos del usuario especificado.

El usuario que ejecuta el procedimiento *SA_SESSION.SET_ACCES_PROFILE* debe tener el privilegio *profile_acces* en la política Oracle Label Security. Además, debe tenerse en cuenta, que el usuario que ha accedido a la base de datos no cambia, si no, que asume las autorizaciones y privilegios del usuario especificado. Por el contrario, el nombre de usuario de Oracle Label Security se cambia.

Este procedimiento es útil para diversas tareas:

- El administrador puede ver el resultado de la autorización y privilegio configurado para un usuario en particular.

- Las aplicaciones que deben tener conexión con cuentas proxy (con el fin de asumir la identidad), como pueden ser aplicaciones de usuarios, a efectos de acceder a datos etiquetados. Con este procedimiento, la cuenta proxy puede actuar en nombre de un usuario de la aplicación.

Sintaxis:

Procedure set_acces_profile (<nombre_política> in varchar2,
<nombre_usuario> in varchar2);

SA_SESSION.SA_USER_NAME: Devolución del nombre de usuario

Esta función devuelve el nombre del usuario actual conectado a la política Oracle Label Security. Así es como se puede determinar la identidad del usuario actual en relación con Oracle Label Security, en lugar del nombre de usuario actual conectado a Oracle.

Sintaxis:

Function sa_user_name (<nombre_política> in varchar2)
return varchar2;

Anexo III: Opciones para la aplicación de políticas Oracle Label Security

Las opciones para la aplicación de políticas de Oracle Label Security pueden personalizarse para cada política. Por ejemplo, una política de Recursos Humanos y una política de Defensa pueden estar en la misma base de datos Oracle y brindar diferentes grados de protección. La aplicación de Recursos Humanos podría utilizar la opción READ CONTROL y la política de Defensa podría utilizar las opciones READ CONTROL y WRITE CONTROL.

READ CONTROL – Aplica la política a todas las consultas; solo las filas autorizadas son accesibles para las operaciones SELECT, UPDATE y DELETE.

INSERT CONTROL – Aplica la política a las operaciones INSERT, de acuerdo con el algoritmo Oracle Label Security para el acceso de escritura.

UPDATE CONTROL – Aplica la política a las operaciones UPDATE en las columnas de datos dentro de un fila, de acuerdo con el algoritmo Oracle Label Security para el acceso de escritura.

DELETE CONTROL – Aplica la política a las operaciones DELETE, de acuerdo con el algoritmo Oracle Label Security para el acceso de escritura.

WRITE CONTROL – Determina la capacidad de INSERTAR, ACTUALIZAR y ELIMINAR (INSERT, UPDATE AND DELETE) los datos de una fila. Si esta opción está configurada, impone INSERT_CONTROL, UPDATE_CONTROL y DELETE_CONTROL.

LABEL DEFAULT – Si el usuario no especifica explícitamente una etiqueta en INSERT, se utiliza el valor *row label (etiqueta de fila)* por defecto de los usuarios. Por defecto, el valor *row label* es internamente computado por Oracle Label Security mediante el uso de valores para autorización de etiquetas especificados para el usuario. Un usuario puede configurar la etiqueta de la fila de manera independiente, pero solo:

En un nivel que sea menor o igual al nivel de la etiqueta de sesión, y mayor o igual al nivel mínimo del usuario.

Para incluir un subgrupo de compartimentos y grupos de la etiqueta de sesión, para lo cual el usuario está autorizado con acceso de escritura.

LABEL UPDATE – Aplica la política a las operaciones UPDATE que establecen o cambian el valor de una etiqueta adjunta a una fila. Los privilegios WRITEUP, WRITEDOWN y WRITEACROSS solo son aplicados si la opción LABEL_UPDATE está configurada.

LABEL CHECK – Aplica la política READ_CONTROL a las sentencias INSERT y UPDATE (INSERTAR Y ACTUALIZAR) para asegurar que la nueva etiqueta de la fila puede ser accedida mediante lectura por cualquier usuario después de una sentencia INSERT o UPDATE.

NO CONTROL – No aplica opciones. No obstante, se puede aplicar una función de etiquetado o un predicado SQL.

Glosario de términos

En el glosario de términos se han incluido, únicamente, los términos más relacionados con el proyecto fin de carrera, evitando la inclusión de términos muy generales.

A

ACL (Acces Control List).- Lista de control de acceso es un concepto de seguridad informática usado para fomentar la separación de privilegios. Su principal objetivo es filtrar el tráfico en equipos de redes, permitiendo o denegando el tráfico de red de acuerdo a alguna condición.

Autenticación.- Proceso por el que una entidad comprueba que otra entidad, denominada principal es quien o lo que dice ser.

Autorización.- Proceso por el que a una entidad de seguridad autenticada se le proporciona acceso a recursos, como a un archivo en el sistema de archivos o a una tabla en una base de datos.

B

Backup.- Copia de seguridad.

Base de datos relacional.- Base de datos que cumple con el modelo relacional, donde a los datos que se almacenan se accede a través de relaciones. Los datos son almacenados en tablas que contienen la información ordenada.

C

Certificados.- Instrucciones firmadas electrónicamente que enlazan el valor de una clave pública con la identidad de la persona, dispositivo o servicio que tiene la clave privada correspondiente.

Certificado X509.- Certificado de clave pública. Es la pieza central de la *infraestructura PKI*, y es la estructura de datos que enlaza la clave pública con los datos que permiten identificar al titular.

Cifrado.- Proceso de convertir datos en un formato que no puede leerse sin una clave especial, por lo que sólo el destinatario previsto puede leer los datos.

D

DBA.- Administrador de una base de datos.

E

Eficacia.- Capacidad de lograr el efecto que se desea o se espera.

Eficiencia.- Capacidad de disponer de alguien o de algo para conseguir un efecto determinado.

Extranet.- Es la parte de la Intranet de una organización que se extiende a usuarios fuera de ella.

I

IEEE.- Instituto de Ingenieros Eléctricos y Electrónicos. Es la mayor asociación internacional sin fines de lucro formada por profesionales de las nuevas tecnologías. Su creación se remonta al año 1884, pero no es hasta 1963, al fusionarse con asociaciones como el AIEE y el IRE, la adopción del nombre IEEE.

Instancia de Oracle.- Una instancia de Oracle está conformada por varios procesos y espacios de memoria compartida que son necesarios para acceder a la información contenida en la base de datos.

Intranet.- Término que describe la implantación de las tecnologías de Internet dentro de una organización, más para su utilización interna que para la conexión externa.

ISO.- Organización Internacional para la Normalización, nacida tras la Segunda Guerra Mundial, es el organismo encargado de promover el desarrollo de normas internacionales de fabricación, comercio y comunicación para todas las ramas industriales a excepción de la eléctrica y la electrónica. Su función principal es la de buscar la estandarización de normas de productos y seguridad para las empresas u organizaciones a nivel internacional.

L

Label default.- Valor por defecto a asociar a una etiqueta de sensibilidad.

Label tag.- Valor asociado al campo que representa a la etiqueta dentro de la tabla de base de datos sobre la que está siendo aplicada la política Oracle Label Security.

LDAP.- (Protocolo Ligero de Acceso a Directorios). Protocolo a nivel de aplicación que permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red. También puede ser considerado una base de datos (aunque su almacenamiento puede ser diferente) a la que pueden realizarse consultas.

Listener.- Proceso servidor que provee la conectividad de red con la base de datos Oracle. El listener está configurado para escuchar la conexión en un puerto específico en el servidor de la base de datos. Cuando se pide una conexión a la base de datos, el listener devuelve información relativa a la conexión. La información de conexión para una instancia de una base de datos provee el nombre de usuario, la contraseña y el SID de la base de datos.

M

Metadatos.- Datos altamente estructurados que describen información, describen el contenido, la calidad, la condición y otras características de los datos. Son en definitiva, información sobre información.

Mínimo privilegio.- Principio según el cual los sujetos deben acceder exclusivamente a aquellos objetos que precisen inexcusablemente para ejecutar sus trabajos o procesos.

O

OLS.- Oracle Label Security.

P

Paquete de software.- Es una serie de programas que se distribuyen conjuntamente. Algunas de las razones suelen ser que el funcionamiento de cada uno complementa a o requiere de otros.

Proceso background.- Proceso que se lleva a cabo en un segundo plano.

S

Script.- Conjunto de instrucciones que permiten la automatización de tareas creando pequeñas utilidades.

Sensitive label.- Etiqueta/s de sensibilidad asociada/s a una política Oracle Label Security. Cuando una política se aplica a una tabla cada uno de los registros de dicha tabla podrá tener una etiqueta de sensibilidad que lo represente.

SID.- Identificador del sistema Oracle, se utiliza para identificar de forma exclusiva una determinada base de datos en un sistema.

SQL.- (Lenguaje de Consulta Estructurado). Es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones en éstas. Una de sus características es el manejo del álgebra y el cálculo relacional permitiendo efectuar consultas con el fin de recuperar, de una forma sencilla, información de interés de una base de datos, así como también hacer cambios sobre ella. Es un lenguaje de cuarta generación.

SSL.- (Secure Sockets Layer). Protocolo diseñado para proveer comunicaciones cifradas en Internet.

T

Tecnología PKI.- Permite a los usuarios autenticarse frente a otros usuarios y usar la información de los certificados de identidad (por ejemplo, las claves públicas de otros usuarios) para cifrar y descifrar mensajes, firmar digitalmente información, garantizar el no repudio de un envío, y otros usos.

Trazabilidad.- La propiedad del resultado de una medida o del valor de un estándar donde este pueda estar relacionado con referencias especificadas, usualmente estándares nacionales o internacionales, a través de una cadena continua de comparaciones todas con incertidumbres especificadas.

Triggers.- (Disparadores). Clase que implementa una interfaz que dispone de un método en el cual se implementa la tarea que debe ser llevada a cabo en caso de que se produzca algún problema, en la mayoría de casos debería deshacer las acciones llevadas a cabo en el método.

Tupla de datos.- Representa un ítem único de datos implícitamente estructurados en una tabla.

U

User Label Authorizations.- (Autorización de Etiquetas de Usuario) Autorización que se ha concedido a un usuario en una política Oracle Label Security.

V

VPD.- (Virtual Private Database) Base de datos virtual privada.

Bibliografía

Libros

- *"ISO 9001:2000 comentada"*, Cianfrani, Charles A.

- *"Oracle 10g: administración y análisis de bases de datos"*, Pérez López Cesar

Documentos electrónicos

- *"Auditing Oracle Security"*, KK Mookhey.

- *"Introduction to Simple Oracle Auditing"*, Pete Finnigan

- *"Norma ISO 9001:2000"*

- *"Norma ISO 9001:2008"*

- *"Norma ISO 90003:2005"*

- *"Oracle Database 10g Security and Identity Management"*, an Oracle white paper, Diciembre 2003.

- *"Oracle Label Security – Mejores Prácticas para Aplicaciones de Gobierno y Defensa"*, Informe Ejecutivo de Oracle, Junio 2007.

- *"Principios de la gestión de la calidad"*

- *"Seguridad en bases de datos: ficciones y fricciones"*. Revista SIC, Noviembre 2006.

- *"Seguridad en Oracle 11g"*

Fuentes electrónicas

- http://es.wikipedia.org/
- http://download.oracle.com/docs/cd/B28359_01/network.111/b28529/toc.htm
- http://www.als-es.com/home.php
- http://www.dataprix.com/es/recopilaci-n-art-culos-oracle
- http://www.monografias.com/
- http://www.oracle.com/technology/deploy/security/index.html
- http://www.oracle.com/technology/global/lad-es/documentation/index.html
- http://www.petefinnigan.com/default/default_password_list.htm
- http://www.rae.es
- http://www.uc3m.es
- http://www.wikilearning.com/